OLTRE 15
PROGRAMMI SU DISCO

EUROPEAN DEVELOPER CONFERENCE

IN COLLABORAZIONE CON AMIGA RESOURCE

ANNO 3 - N. 11
APRILE 1990
L. 14.000
Frs. 21.00

MAGAZINE AMIGA

IL MENSILE JACKSON PER GLI UTENTI DI AMIGA

- Espandere un 500
- Il drive di Amiga
- Citizen Swift 24
- Page Render 3D
- Digi Paint 3
- Gli sprite
- Music-X
- Arp 1.3

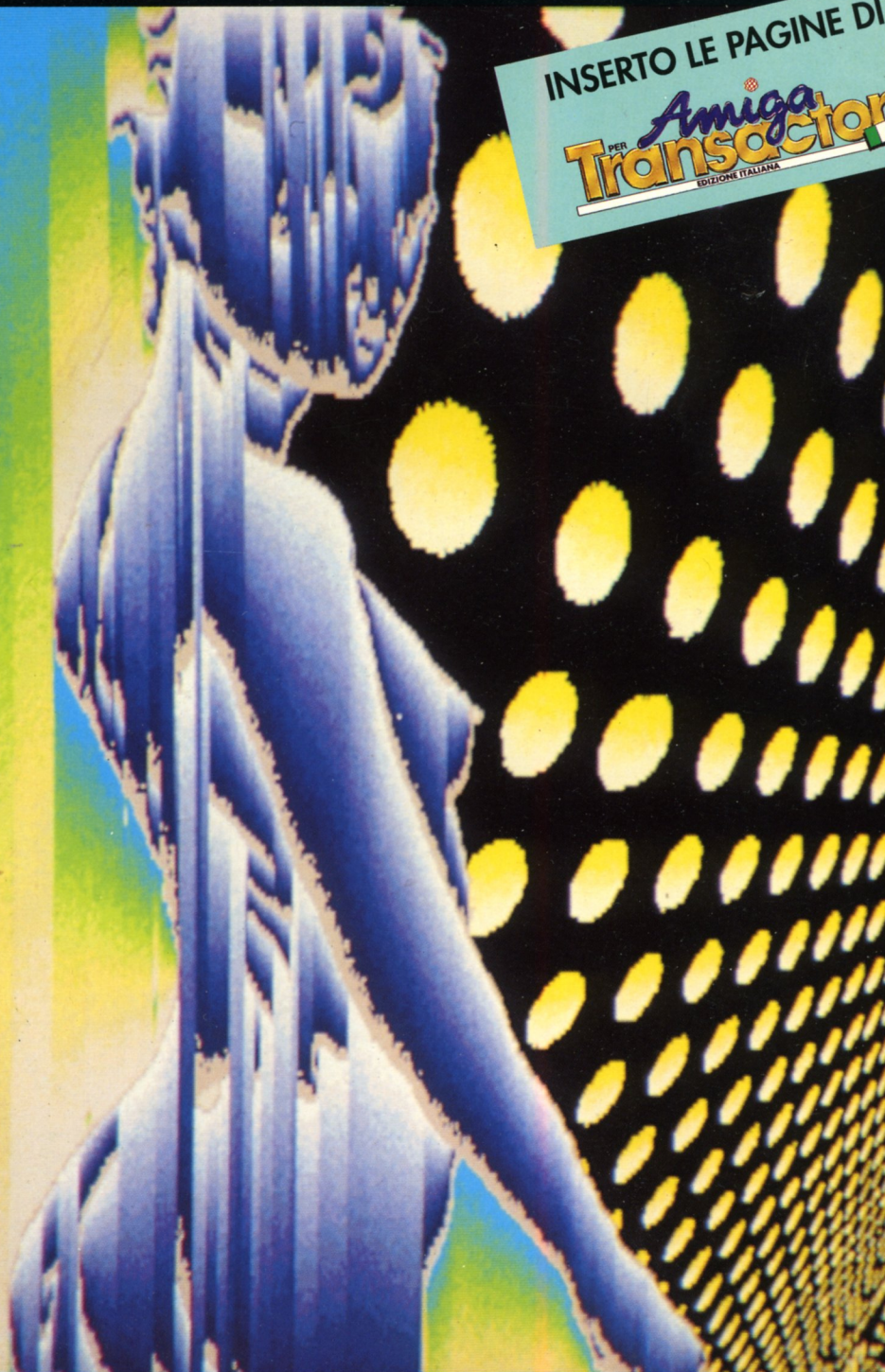
INSERTO LE PAGINE DI
PER Amiga Transactor
EDIZIONE ITALIANA

Spedizione in abbonamento postale Gruppo III/70 Taxe percue (Tassa riscossa) Milano Ferr. Corr.

GRUPPO EDITORIALE JACKSON

Gruppo Editoriale Jackson

Hobby & Home Computer

AMIGA MAGAZINE
ANNO 3 - NUMERO 11
APRILE 1990

Sommario

Editoriale 5

Posta 6
I lettori ci scrivono...

Trends 8
Novità da tutto il mondo

Conferenze 10
European Developer Conference

Il tecnico risponde 12
Il drive di Amiga

Totocalcio 15
TotoAmiga

AmigAvventure 16
E perché non ordinarlo in America ?

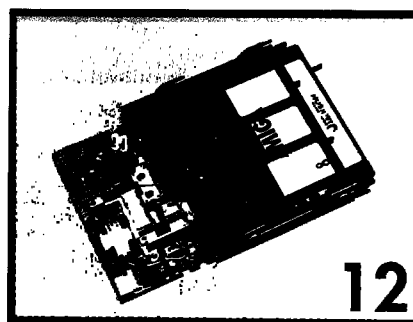
Hardware 19
Citizen Swift 24

Spazio MIDI 22
Music - X

Public Domain 27
ARP 1.3

Grafica 30
Digi Paint 3

ON DISK 33
15 fantastici programmi e...

Le pagine di Transactor per Amiga 35
Un rettifica per boot del 68010
I moduli residenti di Exec
Il Debugging
Interchange File Format

Dossier 53
Espandere un 500

Programmazione Facile 56
Gli Sprite

Grafica 59
Page Render 3D

Top Ten 63

Videogiochi 64

Recensione 71
S.E.U.C.K.

Service 73
Come utilizzare il dischetto

Buyer's Guide 76
Guida all'acquisto: non solo gioco

By David Joiner
Copyright © 1989
MicroIllusions

12

Direttore Responsabile: Paolo Reina
Coordinamento Tecnico e Redazionale: Massimiliano Anticoli
Tel. 02/6948287
Redazione: Stefano Albarelli (On Disk), Romano Tenca (Transactor)
Segreteria di redazione: Elena Ferré - Tel. 02/6948254
Art Director: Marcello Longhini
Copertina, Grafica e Impaginazione elettronica: Cristina Turra
Collaboratori: Lucio Bragagnolo, Cesare Palmieri, Liliana Simonetti, Stefano Riva, Sergio Ruocco, Sebastiano Vigna, Daniele Cassanelli, Mauro Ballocchi, Aldo e Andrea Laus, Gianni Biagini, Renato Acciardi, Stefano Paganini

Group Publisher: Pierantonio Palerma
Direzione Coordinamento Operativo: Graziella Falaguasta
Publisher Area Consumer: Filippo Canavese
Pubblicità: Filippo Canavese
Direzione Sviluppo Pubblicità: Walter Bussolera

SEDE LEGALE
Via P. Mascagni, 14 - 20122 Milano

DIREZIONE - REDAZIONE
Via Pola, 9 - 20124 Milano - Tel.: 02/69481
Fax: 02/6948238 Telex 316213 REINA I

PUBBLICITÀ'
Via Pola, 9 - 20124 Milano - Tel.: 02/6948254
ROMA - LAZIO E CENTRO SUD Via Lago di Tana, 16
00199 Roma
Tel.:06/8380547 - Fax: 06/8380637

INTERNATIONAL MARKETING
Tel.:02/6948233

DIREZIONE AMMINISTRATIVA
Via Rosellini, 12 - 20124 Milano Tel.: 02/69481
Fax: 02/6948238

UFFICIO ABBONAMENTI
Via Rosellini, 12 - 20124 Milano - Fax: 02/6948489
Telex 333436GEJ IT - Tel.: 02/6948490 (nei giorni di martedì, mercoledì, giovedì. 14.30 - 17.30)

Prezzo della rivista: L. 14.000 prezzo arretrato L.28.000
Abbonamento annuo Italia L. 124.000, Estero L. 248.000
I versamenti vanno indirizzati a:
Gruppo Editoriale Jackson SpA
Via Rosellini, 12 - 20124 Milano, mediante l'emissione di assegno bancario o per contanti. L'abbonamento può essere sottoscritto anche utilizzando il c/c postale 11666203

CONSOCIATE ESTERE
GEJ Publishing Group Inc. Los Altos Hills
27910 Roble Blanco
94022 California - Tel.: (001 - 415 - 9492028)
Grupo Editorial Jackson - Calle Alcantara, 57
280016 Madrid - Tel.: 1/4017365

Stampa: F.B.M. (Gorgonzola)
Fotolito: Foligraph (Milano)
Distribuzione: Sodip - Via Zuretti, 25 - 20125 Milano

Il Gruppo Editoriale Jackson è iscritto al Registro Nazionale della stampa al N. 117 Vol. 2 foglio 129 in data 17/8/1982.
Spedizione in abbonamento postale gruppo III/70
Aut.Trib. di Milano n. 102 del 22/2/1988

Il Gruppo Editoriale Jackson possiede per "Amiga Magazine" i diritti di traduzione di **Compute!'s Amiga Resource, Compute! Publications Inc. e Transactor for the Amiga, Transactor U.K.**
Amiga Magazine è una rivista indipendente non connessa alla Commodore Business Machine Inc., né con la Commodore Italiana S.p.a.- C64 e Amiga sono marchi registrati dalla Commodore Business Machine.

Associato al

Mensile associato all'USPI
Unione Stampa Periodica Italiana

Consorzio Stampa Specializzata Tecnica

Testata aderente al C.S.S.T. non soggetta a certificazione obbligatoria per la presenza pubblicitaria inferiore al 10%

Il Gruppo Editoriale Jackson pubblica anche le seguenti riviste:

Amiga Magazine Games - Pc Games - Guida Videogiochi
Supercommodore 64 e 128 - Pc Software - Fare Elettronica
Bit - Computer Grafica & Desktop Publishing - Informatica Oggi
Informatica Oggi Settimanale - NTE Compuscuola
Pc Floppy - Pc Magazine - Automazione Oggi
Trasmissioni Dati e Telecomunicazioni - Elettronica Oggi
EO News settimanale - Strumenti Musicali - Watt
Meccanica Oggi - Strumentazione e Misure Oggi

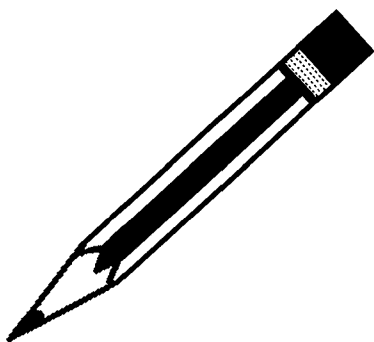
Editoriale

Lo shareware

Su questo numero di Amiga Magazine desideriamo porre all'attenzione dei lettori la filosofia dei programmi distribuiti secondo le modalità del cosidetto shareware, una metodologia distributiva del software che ha conosciuto e conosce tutt'ora grande sviluppo soprattutto negli Stati Uniti.

I programmi shareware sono distribuiti direttamente dai loro Autori tramite banca dati, bullettin board, riviste su supporto magnetico, eccetera; gli Autori stessi incoraggiano il fenomeno della copia fra amici dei loro prodotti, ma ovviamente non per autolesionismo: il concetto di fondo è che essi richiedono, a chi usa il programma, di decidere, in un breve lasso di tempo, se utilizzare effettivamente il programma e, quindi, registrarsi quali utenti autorizzati oppure no.
La registrazione si effettua inviando direttamente agli autori un contributo generalmente molto modesto: il perché di tale impostazione va ricercato nella assenza per il programmatore delle numerose e consistenti voci di costo che caratterizzano le attività di sviluppo di una grande software house e, tutto sommato, l'idea che un numero sufficientemente elevato di adesioni stimolate da un basso prezzo potranno garantire all'autore la necessaria gratificazione economica.

Amiga Magazine intende studiare la realtà dello shareware nel mercato italiano e invitare, comunque, il lettori a considerare con il giusto peso i meriti di chi, senza alcuna certezza di guadagno, ha messo il proprio tempo e il proprio cervello al servizio degli altri.
Meditate gente, meditate...

La redazione

Cercasi Comando

*Spett. Amiga Magazine,
sono uno studente di informatica di 17 anni. Ho notato che sul Workbench 1.3 è presente il comando FF. Potreste spiegarmene il significato? Sicuro di una vostra risposta, porgo distinti saluti.*

Paolo Milani

Il comando FF (FastFonts), prodotto dalla Microsmiths, è un programma che ha lo scopo di velocizzare l'output dei testi sul video. Il risultato non è molto visibile in quanto FF velocizza al massimo del 20% l'output dei testi. La sintassi del comando è:

FF [-0 I -n]

Con FF -0 viene inserito il modo FastFonts (appare il messaggio 'Turning on FastText'), mentre con FF -n questa funzione viene disinserita ('FastText is off'). FastFonts supporta tuttavia un ulteriore parametro non documentato nella maggior parte dei libri sul DOS 1.3:

FF [-0 I -n] [xxxxx.font]

Specificando, infatti, un nome di un file .font, FF sostituisce il nuovo font al posto del topaz (font di default). Il font non deve essere però proporzionale, ma deve essere un Fixed Font, ossia a larghezza fissa. Per ulteriori informazioni sul comando FF e su tutti gli altri dell'AmigaDOS 1.3 ti consigliamo la lettura del libro:

Gruppo Editoriale Jackson
Amiga DOS - Kerkloh, Tornsdorf, Zoller - L. 59.000 (Cod. CC815)

Mac e Amiga

*Spett. Redazione,
ho sentito, da alcuni amici, che è in circolazione un programma simile al Dos-2-Dos che permette di ottenere i file Macintosh in formato Amiga. Potreste gentilmente darmi qualche ulteriore notizia a riguardo?*

Matteo Zappa

Il programma di cui parli si chiama Mac-2-Dos e permette lo scambio di file tra Amiga e Macintosh. Il software ha però bisogno di hardware aggiuntivo, senza il quale non può funzionare.
E' necessaria, infatti, un'interfaccia da collegare ad una porta di Amiga alla quale viene collegato un drive Macintosh.
Quest'ultimo è indispensabile perché i computer Apple utilizzano, durante la lettura e la scrittura dei dati, una diversa velocità di rotazione del disco a seconda della posizione della testina, mentre Amiga mantiene una velocità costante, qualunque sia la posizione della testina di lettura/scrittura.
Mac-2-Dos viene venduto sia con l'interfaccia e il drive, che con la sola interfaccia per i possessori di un drive Macintosh.

Mac-2-Dos è prodotto da:
Central Coast Software
424 Vista Avenue
Albany, NY 12206, U.S.A.

Libri

*Spettabile redazione di Amiga Magazine, sono un assiduo lettore della vostra rivista anche se da poco possessore di un Amiga 500.
Fino ad ora non mi sono interessato molto alla programmazione ma adesso, avendo del tempo libero, ho deciso di avvicinarmi al mondo misterioso di questa stupenda macchina.
I miei quesiti da porvi sarebbero troppi, soprattutto riguardo la programmazione in BASIC e in C, quindi vorrei che mi consigliaste qualche Vostra pubblicazione in merito.
Sicuro dell'aiuto che mi fornirete, ringrazio e porgo distinti saluti.*

Roberto Bompiedi

Ti consigliamo vivamente la lettura dei libri:

- AMIGA Linguaggio C (cod. CL785)
- AMIGA Basic (cod. CL 768)
- AMIGA Tecniche di Programmazione (cod. CC795)

editi dal nostro Gruppo Editoriale.

Prodotti

*Spett.readazione,
ho saputo che la GVP non commercializza solo Hard Disk.
Quale altro componente produce ?
Ringraziando per la risposta, porgo i più cordiali saluti.*

Gianluca Carrettiero

Innanzitutto la GVP (Great Valley Products Inc.) non produce Hard Disk, ma controller per Hard Disk (il controller, in parole povere, è l'hardware che controlla l'interfacciamento tra hard disk e sistema operativo).
Passiamo ora alla tua domanda.
La GVP produce, oltre ai sopracitati controller per HD, anche schede acceleratrici ed espansioni di memoria. La sua più recente produzione è la scheda acceleratrice con microprocessore Motorola 68030 a, tieniti forte, 28 MHz, naturalmente affiancato dall'immancabile coprocessore matematico 68882 (il Memory Management Unit 68852, unità di gestione della memoria, presente sulla scheda A2620 della Commodore non c'è in quanto è "incorporato" nel 68030). La scheda, inoltre, contiene (on board) un controller per Hard Disk e un'espansione di memoria di 4 MByte a 32 bit, in-

dispensabile per 68020 e 68030, espandibili a 8Mb. La scheda per contenere il tutto ha un formato a sandwich (simile alla A2286 della Commodore).
Il prezzo si dovrebbe aggirare intorno ai 4.500.000 lire.

Great Valley Products Inc.
225 PLANK AVE
PAOLI, PA 19301
Tel: 215-889-9411
Fax: 215-889-9416

Precisazioni

Sul numero 9 di Amiga Magazine è apparso, per un errore di stampa, Amiga 25000 UX invece di Amiga 2500 UX.
Ci scusiamo con la Commodore Italiana e con tutti i nostri lettori.

DeskTop Publishing

Spett.Amiga Magazine,
sono da poco possessore di un Amiga 2000.
Vorrei sapere, se esistono e quali sono, i programmi di DTP per Amiga.
Ringraziando anticipatamente per la risposta porgo i migliori saluti.

Bruno Lovati

I programmi di DeskTop Publishing (editoria da tavolo o videoimpaginazione) per Amiga sono molti e diversi tra di loro. Il primo programma di DTP è stato il glorioso PageSetter della Gold Disk che ha praticamente dettato le basi nel mondo Amiga. Va detto che il difetto di questo programma era l'impossibilità di utilizzare stampanti Laser (senza le quali il DTP non ha senso), tanto che la Gold Disk fornì separatamente un'utility per ovviare a questa mancanza. I programmi attuali permettono, invece, di ottenere un risultato di stampa anche su stampanti Laser. I programmi in commercio attualmente sono: Professional Page e PageSetter (Gold Disk), Shakespeare (Infinity Software), City Desk (Sunrize MicroSearch) e Page Stream (Soft- Logik). Per informazioni dettagliate su questi programmi ti rimandiamo al numero 8 di Amiga Magazine, sul quale è apparso uno speciale DTP.

Dalla redazione...

Per ovvie ragioni di spazio non ci è possibile pubblicare tutte le richieste che giungono in redazione.
Cercheremo di rispondere privatamente alle lettere non pubblicate.

International User Group

User Group del mondo unitevi !!

Questo è il motto che ha la nuova rubrica di Amiga Magazine. Se fate parte di un User Group e volete vedere pubblicato il vostro nome, inviateci i vostri dati: nome e il nome del vostro gruppo, indirizzo, telefono Bbs ecc.
Vengono pubblicati anche indirizzi di User Group stranieri in modo da aprire le frontiere fra utenti Amiga. La redazione non si assume nessuna responsabilità sui dati inviati e pubblicati.

Il nostro indirizzo è:
Gruppo Editoriale Jackson
Amiga Magazine - Area Consumer
Rubrica "International User Group"
Via Pola, 9 - 20124 Milano

U.S.A.

Scottsboro Commodore Users Group, Rt. 5, Scottsboro, AL 35768

Montgomery Area Commodore Komputer Society (MACKS), P.O. Box 210126, Montgomery, AL 36121-0126

Amiga/Commodore Club of Mobile, 3868-H Rue Maison, Mobile, AL 36608

Sitka Commodore User's Group, P.O. Box 2204, Sitka, AK 99835

Catalina Commodore Computer Club Inc., P.O. Box 32548, Tucson, Az 85751

Commodore Information Association (CIA), Rt. 1 Box 103 F, Mayflower, AR 72106

Triple-D 64, P.O. Box 301, Reyno, Ar 72462

Los Angeles Amiga Users Group, P.O. Box 947, Culver City, CA 90230

Southern California Commodore Users Group, 11018 E. Rosecrans Ave., Suite 203, Norwalk, CA 90650

Belgio

L'Amiral Club C= 64 & Amiga, P.O. Box 41 B-1090, Brussels, Belgium

Brasile

Commodore Users Group Porto Alegre, Ferreira de Abreu 91/3, 90040 Porto Alegre RS, Brasil

Giappone

Commodore Fan Club, Koji Sugimura, 2-1-10-1107 Higashi-Taishi Yao, Osaka, Japan 581

Spagna

Costa Blanca Computer Club, c/o Ed Kelly, Montebello 25, La Nucia-Alicante, Spain 03530

Svezia

Stockholm Computer Club, Lars Persson, Box 18158, 200 32 Malmo, Sweden

a cura di S. Leemon, M. McLean
(© Compute! Publications, Inc.) e M. Anticoli

20000 leghe sotto i mari

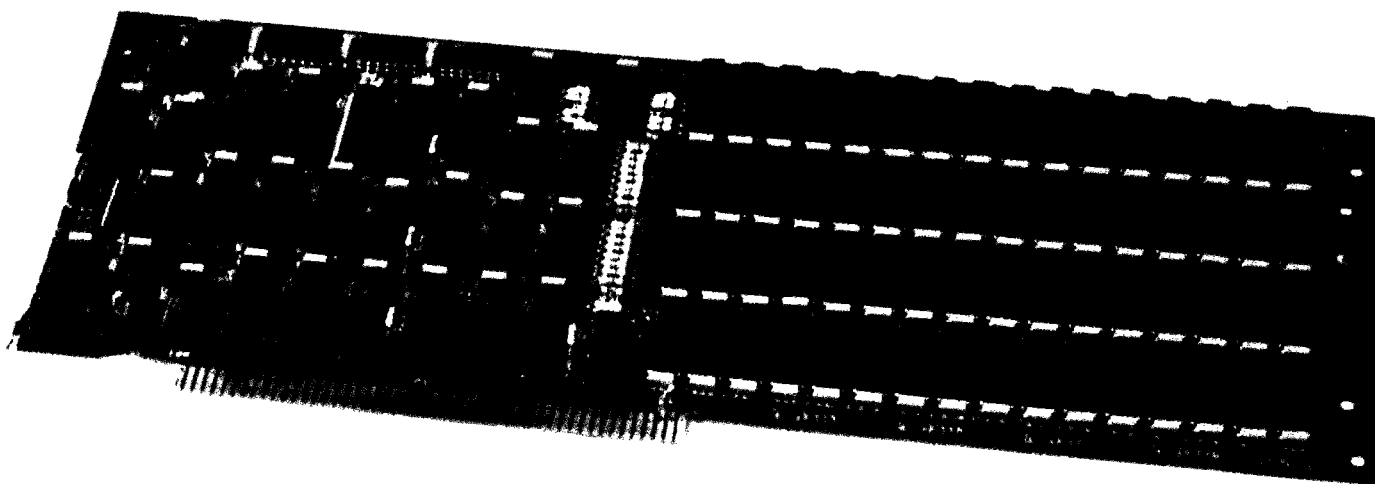
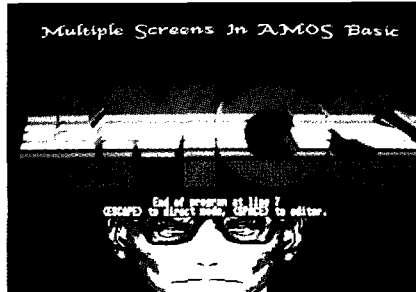
Il gioco che dà il titolo a questa news uscirà con un certo ritardo, poiché il suo programmatore, Jim Sachs, sta collaborando con la Silent Software, il cui gioco Roger Rabbit ha portato a rapporti di lavoro nientemeno che con la Walt Disney.
Il ritardo è quindi giustificato, tanto più che 20000 *Leagues Under The Sea comprenderà molto* probabilmente suoni e immagini digitalizzate fornite dalla Walt Disney stessa.
Fate attenzione ai prossimi annunci della Epyx, probabile distributrice del gioco. ▲

Memorie estese

Megaboard 2000, della Progressive Peripherals & Software, è una espansione di memoria di due Mbyte per Amiga 2000.
Le opzioni a disposizione includono piena autoconfigurabilità e la compatibilità con le routine della versione 1.2 del sistema operativo di Amiga.
La scheda si può acquistare vuota, oppure con uno o due Mbyte di RAM residenti. ▲

Il Computer via etere

Da Ottobre va in onda su ANTENNATRE "TUTTOCOMPUTER GIOVANI", una trasmissione dedicata alle nuove schiere di appassionati del computer.
Ogni Mercoledì dalle 17,45 al 18, fino al prossimo Giugno, gli "informatici" che vorranno sintonizzarsi su *questa emittente lombarda, avranno modo di* conoscere, in diretta, le novità del settore.
"TUTTOCOMPUTER GIOVANI" fa parte di "ANGOLI", una diretta televisiva condotta da Raffaella Corradini.
Lo spazio dedicato all'informatica è presentato da quattro ragazzi: Riccardo Panza, 15 anni, frequenta la 3ª Liceo Scientifico di Pilotello; Stefano Serafin, 17 anni, al 4° anno di informatica; Corrado Toscano e Stefano Beretta, 18 anni, che frequentano il 5° anno di informatica, questi ultimi tre sono studenti dell'ITIS Galvani di Milano. Le puntate si articolano su diversi argomenti che fra i giovani vanno per la maggiore: videogiochi, news, hardware e robotica.
Prossimamente saranno coinvolti anche studenti e studentesse che hanno sviluppato del software di particolare interesse, non solo per quanto riguarda i giochi, ma anche per quanto riguarda la didattica e le applicazioni professionali.
Gli esecutori dei progetti scelti saranno invitati presso l'emittente e spiegheranno personalmente ai telespettatori la loro applicazione. ▲

Tuttocomputer Giovani - Antennatre
Via Per Busto, 15 - 20025 Legnano (MI)

Amos

La Mandarin Software vi offre, con AMOS, la possibilità di progettare giochi per mezzo di semplici comandi BASIC, il che rende il programma molto più flessibile dei package di "Construction Kit". Oltre alle istruzioni BASIC tradizionali, ve ne sono di speciali, molto potenti, che velocizzano di tre volte almeno l'esecuzione.
AMOS è la versione completamente riscritta di STOS, il package best-seller studiato per Atari ST e come lui, non si riduce esclusivamente ad un tool per la creazione di giochi, ma può sviluppare con la stessa facilità e velocità programmi educativi, database, pacchetti finanziari eccetera. ▲

User Group

Nel mese di Febbraio è stato fondato il P.I.S.A. (Programmatori Italiani per lo Sviluppo di Amiga), che è subito stato accettato tra i Commodore Certified Developers e gode del programma di supporto agli sviluppatori della Commodore Italia. Il P.I.S.A. è un gruppo di Programmatori il cui scopo è lo studio di Amiga e la divulgazione di notizie poco referenziate. I progetti sono quindi essenzialmente didattici. I progetti a cui stanno lavorando i vari gruppi di interesse sono:

- Una scheda di Input/Output Analogico e Digitale Modulare con attivazione delle procedure di Autoconfig.

- Un rivoluzionario programma di BBS, con interfaccia software diretta con Amiga, che gestisca direttamente le risorse della macchina.

- Una originale rete locale a basso costo comprensiva di software intelligente.

Il P.I.S.A. U.G. può contare su molti collaboratori più o meno stretti, tutti collegati con il mondo universitario e della ricerca.

Per ulteriori informazioni contattare:
P.I.S.A. U.G.
c/o Emmanuele "Exedre" Somma
Via S.Marta, 57 - 56127 Pisa - Italia

oppure sul nodo 2:332/23.32 di Fidonet spedendo le proprie richieste a PISA USERGROUP. ▲

Amiga '90

Dal 9 al 12 Maggio si terrà a Basilea (Svizzera) Amiga '90, la più grande mostra europea dedicata al computer Amiga.
Saranno presenti più di 100 espositori provenienti da tutto il mondo. ▲

Custodian II

Tricom, società di sistemi di comunicazione dati, ha introdotto sul mercato Custodian II, un sistema di sicurezza, concepito per combattere la crescita del delitto computerizzato.
Oggigiorno la comunità degli affari si affida all'elaboratore per l'efficacia delle sue operazioni.
Il problema è che gli elaboratori non sono del tutto affidabili. Consegnano le loro informazioni attraverso linee telefoniche a chiunque abbia la parola d'ordine giusta. Permettono anche agli intrusi (chiamati "hacker") di inserire i virus che possono scombussolare un'intera rete.
Il sistema sviluppato dalla Tricom viene collocato tra la porta dell'elaboratore e il modem connesso alla linea telefonica ed è progettato per bloccare l'hacker, mentre rimane amichevole con gli utenti autorizzati.
Il Custodian II è in grado di provvedere sino a dieci livelli diversi di sicurezza che vanno dalla protezione semplice di una parola d'ordine, fino a un sistema di carta d'identificazione personale nominato "Identicard".
Custodian II può essere utilizzato anche per fornire un'ulteriore sicurezza nella richiamata, dove utenti non autorizzati possono essere identificati velocemente richiamando il numero telefonico del possessore della parola d'ordine. "Identicard" è un generatore digitale, della misura di una carta di credito, che utilizza una tecnica di sincronizzazione che aggiorna un numero di sei cifre sulla carta d'identificazione automaticamente ogni 60 secondi.
Quindi, l'"Identicard" fornisce il massimo livello di sicurezza per un gruppo crescente di utenti che hanno bisogno di accesso alle reti telefoniche da una automobile o da un cantiere di un cliente dove non può funzionare il metodo di richiamata.
Per una maggiore sicurezza, un sistema di registrazione documenta ogni volta che il sistema viene usato, o direttamente in tempo reale su una stampante, o su un dispositivo da processare in un secondo tempo. ▲

European Developer Conference

Sebastiano Vigna

A Parigi, dal 7 al 9 Febbraio, si è svolto il terzo congresso europeo per gli sviluppatori di hardware e software per Amiga: ovviamente non potevo mancare.
E' uno dei punti chiave del piano di supporto Commodore per gli sviluppatori, e il momento in cui vengono in genere mostrati nuovi prodotti in via di completameno. Premetto che ad ogni partecipante è stato fatto firmare un "non-disclosure agreement", vale a dire un pezzo di carta che impegna a versare almeno $10.000 nel caso venissero rese pubbliche le conoscenze acquisite senza il permesso della Commodore. Quel che potrò dire sarà, comunque, solidamente documentato, contrariamente a molti pettegolezzi circolanti negli ambienti Amiga, che causano più danno che altro.
E d'altra parte, il lavoro di ristrutturazione, riprogrammazione e make-up a cui la Commodore sta sottoponendo il software di sistema è così vasto (e lo testimonia un ENORME plico di documentazione) da meritare, in futuro, un discorso più approfondito.

Alla conferenza erano presenti quasi tutti i progettisti Commodore, capeggiati da Andy Finkel (il responsabile del settore software). L'unica grossa assenza era Jim Mackraz, uno degli "originali". In compenso, Jay Miner, il designer dei chip custom di Amiga, era presente alla manifestazione. L'ho visto aggirarsi per tre giorni, silenzioso e intento a osservare il grande interesse suscitato dalla macchina da lui progettata.
Il vedere centinaia (sì, eravamo proprio tanti) di sviluppatori dediti ad Amiga penso gli abbia fatto un enorme piacere.

Il pubblico era composto fondamentalmente da europei, sebbene un paio di dozzine di americani fosse presente, assieme a un australiano e a un neozelandese.
I francesi, ovviamente, erano ben rappresentati, e così anche i tedeschi, mentre il gruppo degli italiani era costituito da una quindicina di persone.

Il congresso era articolato in diverse conferenze tenute in parte dalle persone attualmente al lavoro sul software di sistema di Amiga e in parte da esterni, in particolare Bill Hawes e John Toebes.
Il primo è il creatore di ARexx, di ConMan e di WShell, il secondo è il fondatore della Software Distillery, ma è ben più noto per aver curato le release 4.0 e 5.0 del compilatore Lattice, e, soprattutto, per averlo portato a livelli di eccellenza tali da farlo utilizzare alla Commodore: tutti gli esempi del nuovissimo Rom Kernel Manual: Libraries and Devices sono scritti proprio in Lattice 5.04. La battaglia, comunque, continua, dato che la Manx ha appena messo in commercio, dopo due anni di "ibernazione", la versione 5.0.

I due personaggi sono uno l'opposto dell'altro. Bill Hawes è il prototipo del simpatico timido, nonostante i suoi trentasette anni e una fama delle migliori in ambiente Amiga. John Toebes, invece, è un giovanissimo rampante, molto sicuro di sé e assertivo nella comunicazione. Se come me, ve lo aspettavate quarantenne o giù di lì, vi siete sbagliati di grosso; infatti, John ha solo 22 anni.
Bill Hawes ha, ovviamente, tenuto una conferenza su come far comunicare ARexx con i propri programmi, mentre John Toebes ha descritto diverse tecniche di programmazione moderne che permettono di accelerare lo sviluppo del software. Magari ne parleremo in altra sede.

La migliore notizia che ho ricevuto è che, alla fine, ARP sarà nella 1.4. Non ci sarà una "arp.library", ma Charlie Heath ha lavorato con i programmatori Commodore alla riscrittura del DOS, e, quasi, tutte le funzionalità originariamente in ARP sono state infilate nelle varie parti del sistema. Sembra che finalmente potremo scrivere e leggere un file senza dover andare a farci un caffé nel frattempo.

Una delle conferenze di maggiore interesse, per chi scrive software per Amiga in Italia, è stata quella sul piano di supporto sviluppatori. In parte sono state ribadite le direttive dello scorso DevCon, ma ci sono parecchie novità degne di nota.

Ci si può iscrivere a tre livelli, e cioè come sviluppatore Commerciale, Certificato o Registrato. Il livello più alto è quello Commerciale ($450 annui), che dà accesso alle versioni preliminari dell'hardware e del software di sistema, nonché a forti sconti sulla documentazione.
Per registrarsi come commerciali, occorre l'approvazione della Commodore, che, invece, non è richiesta per i certificati o per i registrati. Mentre per i primi ($250 annui) sono disponibili una serie di servizi in comune ai commerciali, come l'ac-

cesso alla conferenza chiusa degli sviluppatori su BIX, i registrati sembrano dover pagare $125 solo per ricevere AmigaMail e accedere alla ADSPE BBS.

ADSPE significa Amiga Developers Support Program Europe, e la ADSPE BBS è un nuovo servizio che la Commodore ha intenzione di mettere a disposizione degli sviluppatori. Visto che l'assistenza telefonica è dispendiosa e costituisce uno spreco di informazioni (bisogna ripetere dieci spiegazioni identiche a dieci persone diverse), verrà reso disponibile un BBS a conferenze ove potranno essere poste domande e risposte in maniera più efficiente e organizzata.
Il software di comunicazione utilizzato sarà UNIX, e qui entrerà in gioco la versione Amiga di UUCP (UNIX to UNIX copy) scritta da Matt Dillon, che permette di dialogare alla UseNet da AmigaDOS. Per cercare, inoltre, di fornire un'assistenza simile a quella fornita da BIX negli USA, dove la Commodore tiene due conferenze per gli sviluppatori, la Commodore progetta di supportare gli sviluppatori europei attraverso GEnie, il sistema di conferenze della General Electric che sta cominciando ad aprire porte d'ingresso (commutate) anche qui in Europa. In questo modo, anche chi non ha accesso ad una rete a pacchetto (leggi ITAPAC, dato che in Italia ce n'è purtroppo una sola) dovrebbe poter accedere facilmente a questo servizio.

Si vengono automaticamente a porre a questo punto alcune domande piuttosto spinose. La prima: BIX è un servizio flatfee, ovvero con $39 ogni trimestre si ha a disposizione un tempo d'accesso illimitato. GEnie, per di più chiamato da una località europea, verrebbe a costare almeno una dozzina di dollari l'ora, più le spese telefoniche per il collegamento alla porta d'ingresso. A queste condizioni, i programmatori d'oltreoceano sarebbero enormemente favoriti; vi garantisco che non è facile ragionare con calma quando si pensa che ogni minuto vanno via tot lire. Il secondo problema è che lo stabilire una "base" di supporto su GEnie implicherebbe separare gli sviluppatori americani da quelli europei, con tutti i problemi di confronto, competizione e spreco del caso. E se c'è un punto fondamentale per il futuro di Amiga, è proprio una più stretta collaborazione tra europei e americani.

Al di là di queste critiche, lo sforzo fatto dalla Commodore è ammirevole: e gli oneri che essa stessa si è sobbarcata, soprattutto nel caso degli sviluppatori commerciali, sono notevoli e mostrano come il rimescolamento a livello dirigenziale abbia "partorito" una politica commerciale molto più dinamica ed efficiente.
Chiunque volesse saperne di più sul piano di sviluppo, non ha che da telefonare alla Commodore Italiana.

Tra una conferenza e l'altra ho avuto tra l'altro occasione di incontrare Tom Rokicki, autore di tonnellate di PD per Amiga, nonché di AmigaTeX e della documentazione sul blitter presente nell'Hardware Manual. Dopo aver parecchio insistito per convincerlo che in Italia eravamo proprio sessanta milioni (non ci voleva credere, su un pezzetto così piccolo non ci poteva stare tanta gente...) gli ho chiesto se oltre a me c'erano altri utenti di AmigaTeX in Italia: e qui ho dovuto incassare una risposta che, benché assolutamente non malevola, mi ha causato un certo dispiacere: "Sì, ce ne sono alcuni altri, ma in generale preferisco non mandare AmigaTeX in Italia perché non c'è alcun copyright e la gente duplica a piacere". Se vi siete mai chiesti come ci vedono all'estero... E d'altra parte, è proprio vero: un paio di redattori di una rivista italiana sono stati strapazzati da Perry Kolowitz della ASDG in quanto, pur avendo recensito CygnusEd 2, non l'avevano ottenuto in modo legale. Quello che mi lascia un po' di amaro in bocca è che, pur essendo io utente registrato di CED, a causa di qualche incomprensione su carte di credito e simili, non ho ancora ricevuto la mia upgrade, e non ho quindi potuto offrirvi una recensione "seria".
A volte essere onesti può costare parecchio in fatto di tempi editoriali; non ho l'uso di scagliarmi ferocemente contro la pirateria, ma questo fatto mi ha veramente infastidito.

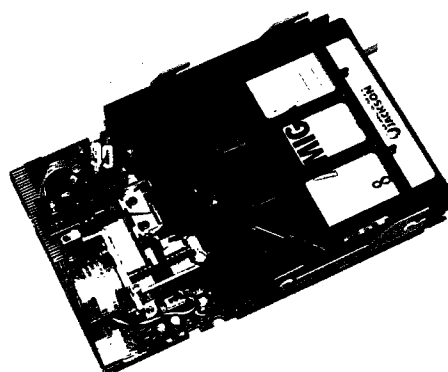
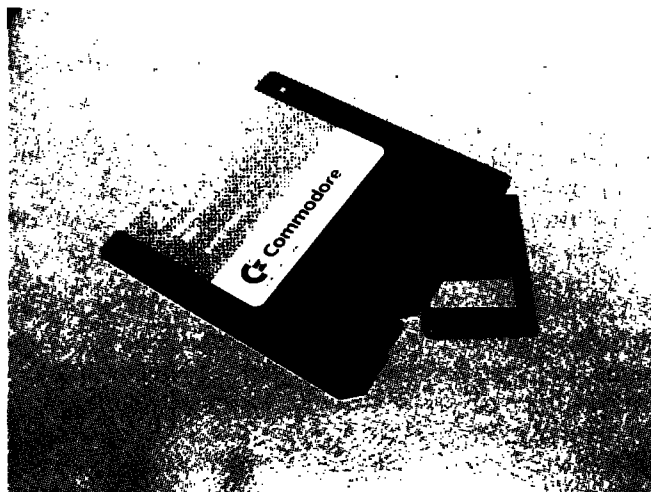
Durante le ultime battute del congresso diversi produttori di hardware e software hanno mostrato i loro prodotti.
Una notizia freschissima è l'uscita della 5.05 della Lattice (l'ultimo giorno del congresso John Toebes ha detto "dovrebbe essere uscita ieri", quindi fate un po' i conti). Fa sempre piacere vedere un ottimo prodotto, ma fa ancora più piacere vederlo ben supportato e aggiornato.
Alcuni tedeschi mostravano, invece, un emulatore Atari ST perfettamente funzionante, di cui non capisco però l'utilità, a parte farlo girare con un 68030 per avere un turbo ST.

Non si può fare a meno di tornare un po' troppo ottimisti e sovraeccitati da questo tipo di incontri, ma l'impressione generale è che Amiga stia prendendo il volo.

L'anno a venire sarà ricco di novità sia nell'hardware che nel software, e probabilmente vedrà una nuova generazione di prodotti, supportati dai miglioramenti al sistema operativo. Non ci resta che aspettare, pazientemente, la 1.4. ▲

Il drive di Amiga

a cura di Computer Lab
V.le Monte Nero, 66 Milano

Che cos'è un drive? Un drive è un dispositivo che permette il flusso di informazioni da e per il computer, inteso come CPU; cioè permette al software di interagire con l'hardware per l'elaborazione di determinate funzioni. Quanta più importanza riveste un particolare apparato tanto più occorre conoscerne la struttura, il funzionamento, i punti deboli per evitare di arrecare danni con i conseguenti effetti negativi che la maggior parte di voi conosce. Per capire il funzionamento dei drive è necessario conoscere il disco, inteso come supporto magnetico. Allo scopo prendete un dischetto meglio se non più utilizzabile, smontatelo e osservatene le diverse componenti: la custodia, parte in plastica dalle dimensioni di 90 x 94 mm con spessore di 3,3 mm, comprende una parte in metallo scorrevole che funge da chiusura della finestra attraverso la quale le testine entrano in contatto con il supporto magnetico.
Tale protezione viene tenuta aperta dal drive solo a disco completamente inserito e una molletta inserita all'interno la richiude al momento dell'estrazione. Il supporto magnetico è costituito da un disco flessibile ricoperto da uno speciale strato magneticamente sensibile, assicurato ad un disco metallico che ne permette l'appoggio al drive. La parte magnetica è composta di particelle orientabili che assumono carica magnetica positiva o negativa in funzione del segnale di scrittura. L'utilizzo del dischetto è determinato dal drive; la prima operazione da eseguire su dischetto vergine è la formattazione. In questa fase le testine del drive, grazie ad una corrente orientano le particelle magnetiche creando così una suddivisione in tracce (80) e settori preparando il dischetto ad accogliere, nel caso di Amiga, 880 Kb di informazioni. Tutte queste informazioni relative al numero di TRK e settori vengono scritte in TRK 0. Le tracce sono piste circolari, la 0 è la più esterna mentre i settori sono porzioni di traccia. In fase di scrittura le testine leggono la TRK 0, identificano le TRK e i settori liberi, scrivono i dati e costruiscono in TRK 0 una sorta di indice con i nomi dei file. In fase di lettura avviene il contrario: nelle testine scorre una corrente che viene modificata per effetto della variazione del campo magnetico creato dalle particelle + e - del dischetto. Si tenga presente che nei drive, a differenza degli Hard disk, le testine poggiano fisicamente sul disco che viene ruotato a circa 300 giri/minuto solo in fase di lettura o scrittura. Negli Hard Disk, invece, la velocità è maggiore e le testine restano sollevate a qualche micron dai dischi. Ritornando al drive possiamo dire che è composto di due parti: una meccanica, che contiene un motore per la rotazione del disco e uno per lo spostamento del gruppo testine (questo secondo motore del tipo step by step genera lo spostamento assiale grazie ad un perno elicoloidale). Vi è poi una guida che accoglie il dischetto che esegue due

Signal name	I/O	Pin Number	Pin Number	Signal name
Disk Change	Output signal	2	1	GND
(in use)	Input signal	4	3	GND
Drive select 3	Input signal	6	5	GND
Index	Output signal	8	7	GND
Drive Select 0	Input signal	10	9	GND
Drive Select 1	Input signal	12	11	GND
Drive Select 2	Input signal	14	13	GND
Motor On	Input signal	16	15	GND
Direction Select	input signal	18	17	GND
Step	Input signal	20	19	GND
Write Data	Input signal	22	21	GND
Write Gate	Input signal	24	23	GND
Track 00	Output signal	26	25	GND
Write Protect	Output signal	28	27	GND
Read Data	Output signal	30	29	GND
Side Select	Input signal	32	31	GND
Ready	Output signal	34	33	GND

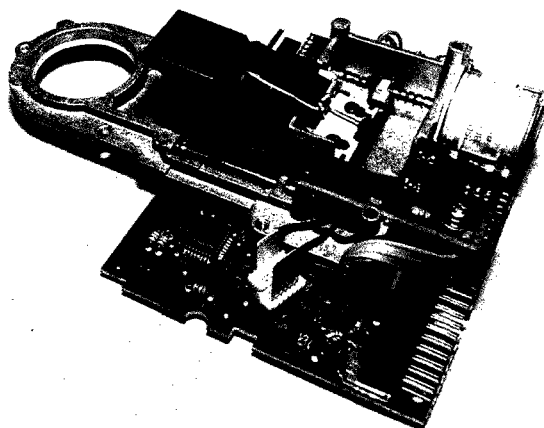
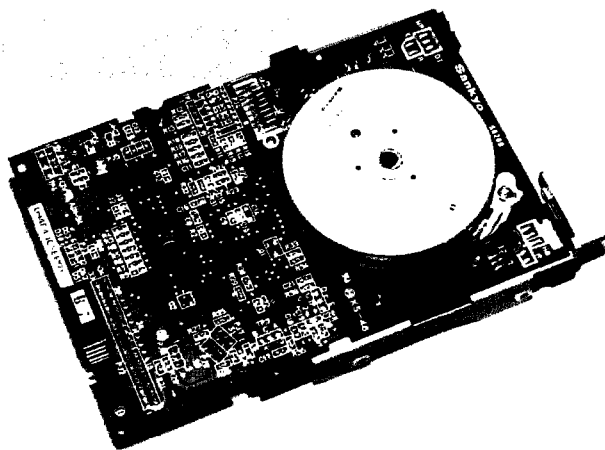
funzioni principali: apertura della finestrella con contemporaneo abbassamento del dischetto e sistemazione delle due testine; una volta inserito il disco un microswitch controlla se il dischetto è o meno protetto in scrittura. L'altra parte del drive, quella elettronica o logica ha funzioni di gestione del drive e dialogo con CPU; qui le linee di comunicazione sono di due tipi, una power costituita da due alimentazioni continue a 5 e 12 Volt e l'altra dei dati comprendente 17 segnali (vedasi tabella). Come interagiscano fra loro i diversi segnali in relazione al drive è subito detto: alcuni mantegono in READY l'unità fino a che siano rispettate determinate condizioni tipo l'INDEX, che controlla le velocità del disco in stretta relazione con il motor on; ogni impulso di STEP muove di una traccia le testine mentre il segnale WRITE PROTECT conferma o meno la possibilità di scrivere sul disco. Si tenga presente che tutti questi controlli vengono eseguiti in tempi dell'ordine dei millisecondi e nanosecondi. Risulta, alla luce di quanto descritto, abbastanza semplice comprendere come un drive per l'elevato numero di operazioni che compie, sia un'unità molto complessa alla quale va prestata molta attenzione (anche in fase di riparazione). Provate, per esempio a considerare cosa capita estraendo il dischetto durante la fase di lettura e scrittura (led acceso). Questa non corretta operazione comporta un conseguente blocco logico con una notevole sollecitazione meccanica: si tratta cioè di far alzare immediatamente le testine mentre compiono uno spostamento sul disco che sta girando a 300 giri al minuto. Fin qui abbiamo trattato l'argomento drive teoricamente, ma in pratica quali sono i difetti più ricorrenti?
Analizziamo i dati relativi alle riparazioni effettuate su Amiga e vediamo subito quale è l'incidenza di intervento su questo device: negli Amiga 500 su 100 apparecchi che entrano in laboratorio per riparazione circa 30 presentano difetti connessi al drive; per gli Amiga 2000 il tasso scende, mentre per gli Amiga 1000 sale intorno al 50%. Perché queste differenze? Amiga 1000 risente molto del fattore utilizzo in quanto ha visto la sua nascita circa 4 anni fa, Amiga 2000 è ancora destinata ad un'utenza più preparata, attenta, mentre Amiga 500 ha sostituito in molti casi il C 64 con drive e l'utente che in molti casi non è a conoscenza del funzionamento del drive lo utilizza in modo improprio. Si tenga, infine, presente che sugli Amiga sono montati cinque tipi di drive prodotti con soluzioni tecnologiche differenti orientate a migliorare le prestazioni che a volte manifestano comportamenti diversi con gli stessi programmi. Un esempio per tutti: il drive MATSUSHITA tipo JU 253 riconoscibile anche per la maggior rumorosità, presenta problemi nel caricamento di alcuni programmi perché, secondo nostra convinzione, il motorino dedicato allo spostamento della testina, controllato elettronicamente, risulta essere leggermente più lento. Quando il programma in fase di caricamento richiede spostamenti "lunghi" del gruppo testine perde il controllo con conseguente messaggio di errore e/o apparizione del GURU. Non si tratta in questo caso di un guasto dell'hardware bensì il produttore di software ha sviluppato il proprio progetto lavorando sulle caratteristiche di altri drive senza tener conto di incontrare, nel tempo, drive perfettamete nella norma ma con leggere differenze. Numerosi utenti Amiga 500 si sono presentati al nostro laboratorio lamentando questo tipo di problematiche. Per tutti loro, purtroppo, la soluzione è di non utilizzare quel software (circa 5 pacchetti) sopportando di vederlo funzionare su Amiga dell'amico. Altre richieste di intervento sono motivate da caricamenti irregolari: molti di questi casi sono stati risolti con la taratura delle testine aumentando così il livello del segnale in lettura. Il disallineamento delle testine è dovuto principalmente a due fattori: tipo di drive e utilizzo di programmi particolari che generano frequenti passaggi dalle trac-

ce interne a quelle esterne. Un discreto numero di interventi si conclude con la sostituzione della parte elettronica, ma in molti casi si rende necessaria la completa sostituzione del drive. Fatti curiosi ne abbiamo visti come, per esempio, carta di caramelle introdotta in chissà quale modo dal fratellino più piccolo, fiori secchi o altri "corpi estranei" finiti misteriosamente all'interno del drive. Riguardo alla varietà dei corpi estranei che provocano a volte seri problemi al drive è doveroso segnalare l'aumento dei danni causati dai dischetti che per motivi diversi, generalmente per la qualità, perdono la parte metallica e relativa molletta; al momento dell'estrazione del dischetto la testina superiore resta agganciata e viene "strappata" verso l'esterno. In questi casi l'unico intervento possibile è la sostituzione completa del drive e ovviamente non può essere considerata la garanzia fornita da Commodore Italiana. Discorso a parte merita Amiga 2000.

Amiga 2000 è stato progettato come prodotto "aperto", espandibile e questo fatto ha autorizzato tutti a metterci le mani. Installare il 2° drive è un'operazione semplice per chi sa distinguere, ad esempio, il pin 1 dal pin 34 del connettore dati.

Non sono pochi gli utenti A 2000 che hanno richiesto l'intervento dopo aver tentato l'installazione del 2° drive: alimentazioni invertite, il che significa alimentare le linee da 5V con 12V e viceversa, oppure connettori dati danneggiati ecc. Gli aspetti più curiosi si sono visti per porre rimedio alla mancanza dei distanziatori necessari a fissare il drive; dal pezzo di legno al bambù o al bullone dell'11 c'è solo l'imbarazzo della scelta!

Mettetevi solo per un attimo dalla nostra parte: laboratorio autorizzato Commodore Italiana...

Si assiste peraltro ad un secondo fenomeno, più grave perché motivato da lavoro eseguito ad arte: al momento dell'acquisto di un A 2000 generalmente viene richiesto come optional il 2° drive lasciando "l'onere" al negoziante dell'installazione. Ebbene, non è infrequente il caso in cui il drive originale venga sostituito con altro a basso costo uguale al 2° con il risultato di avere due drive anziché uno di pessima qualità e basso affidamento.

In questo caso poi il cliente subisce un secondo danno difficilmente valutabile rappresentato dalla scadenza della garanzia. Ci sia concesso concludere con alcuni suggerimenti primo fra tutti quello di utilizzare il computer nel modo più corretto attenendosi alle informazioni del manualetto.

Secondo aspetto da non sottovalutare visto il valore dell'investimento è di cercare un rivenditore di fiducia non dedito a pratiche del tipo sopra riportate.

Terzo è un invito: in caso di malfunzionamento del vostro Amiga prima di affidarlo a mani inesperte richiedete l'intervento al laboratorio autorizzato Commodore.

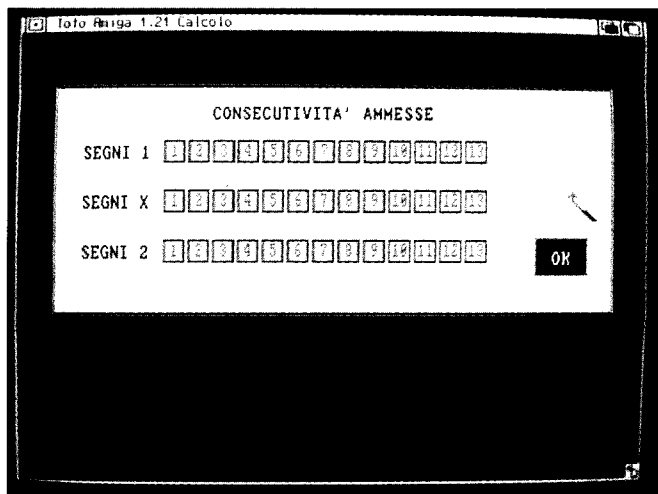
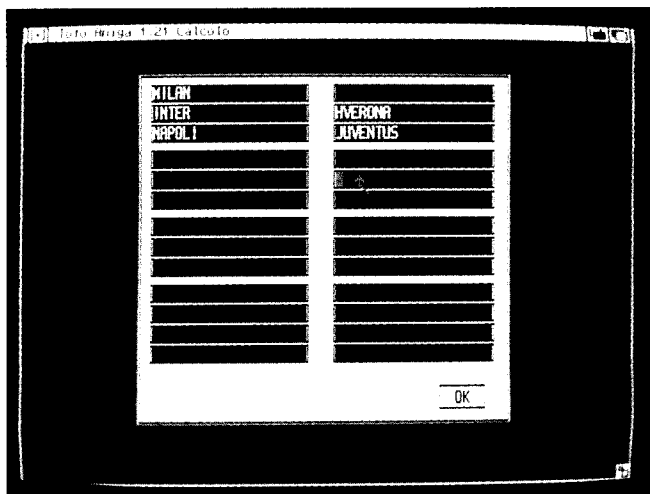
Per quanto ci riguarda, ormai ci conoscete, utilizziamo procedure di test affidabili e diamo informazioni sul tipo di intervento eseguito. Per quanto concerne l'installazione di drive o altri add-on ricordate che forniamo anche il solo servizio di installazione nel caso in cui il materiale aggiuntivo sia già stato acquistato. Un motivo in più per lavorare tranquilli con il vostro Amiga? La nostra esperienza di riparazione conta circa 3000 Amiga riparati. ▲

TOTOAMIGA: Come vincere al Totocalcio, Enalotto e Totip

Presentiamo pregi e difetti, pochi, dell'ultimo programma, per concorsi a premi, proposto dal Gruppo Editoriale Jackson.

Claudio Grassi

Le ultime vincite miliardarie al totocalcio e i recenti record di montepremi, indicano come sia sempre più vasto l'interesse della gente nei confronti di questi concorsi a premi. Non c'è quindi da stupirsi se un sempre più largo numero di persone si dedica all'utilizzo di sistemi sempre più sofisticati ricercando potenza e velocità di calcolo ed esigendo, soprattutto, agilità e facilità nelle operazioni di stesura del pronostico in cui riversare la speranza di vincite favolose.
L'avvento dei computer ha indirizzato l'interesse di molti utenti alla ricerca di programmi idonei a svolgere con semplicità e rapidità tali compiti. Il mercato del software, sempre attento alle esigenze dell'utente, ha sviluppato programmi di vario tipo per soddisfare le richieste degli eventuali acquirenti e TOTOAMIGA risulta essere l'ultimo prodotto che si presenta in commercio. Viene pubblicizzato come il più evoluto per lo sviluppo di pronostici e concorsi e in effetti mantiene molte delle sue promesse. Può sembrare facile, del resto, visto che la concorrenza nell'ambito di programmi di tale genere, implementati per macchine della serie Amiga, è assai limitata. Ma se lo paragoniamo ad altri programmi, e sono numerosi, dedicati a computer più potenti e, se vogliamo, specifici, (PC IBM e compatibili), possiamo constatare che le sue caratteristiche lo rendono competitivo sotto molti punti di vista; prima fra tutte la facilità di impiego. Nella schermata di presentazione sono rappresentate tutte le funzioni disponibili e l'uso del mouse rende estremamente agibile l'ingresso ai vari sottomenu e, ad un esperto di sistemistica, non è nemmeno necessario l'utilizzo del manuale tanto ne è intuitivo l'uso. Le opzioni sono molteplici: dalle più semplici e necessarie quali la riduzione, la consecutività dei segni, il condizionamento delle colonne, le percentuali di probabilità, per fare qualche esempio.

(segue a pag. 18)

E perché non ordinarlo in America?

Sebastiano Vigna

Dopo quasi quattro anni di convivenza con un Amiga (prima un 1000, poi un 2000, poi ancora un 2000 6.0) tre mesi fa avevo deciso che era il caso di comperare un disco rigido. Contrariamente ai primi tempi bui, in cui era già tanto sapere che esistevano gli hard disk, figuriamoci comprarli, si era ormai diffusa una "cultura rigida", e l'idea che uno slot del mio 2000 potesse essere occupato da un bel controller non mi sembrava più tanto fantascientifica.
Ovviamente, data l'entità della spesa era il caso di documentarsi bene. Dopo aver seguito per diverse settimane varie riviste americane, letto decine di pubblicità e recensito un hard disk importato in Italia, presi la mia decisione: volevo il disco rigido più veloce della Terra.
La scelta, a questo punto, era obbligata: per quanto riguarda l'elettronica, il controller HardFrame 2000 della MicroBotics, con la cui espansione per il 1000 avevo felicemente convissuto per parecchio tempo; e per quanto riguarda la meccanica, un bel Quantum ProDrive con tempo d'accesso medio di 11ms, dello stesso tipo (ma non ovviamente delle stesse dimensioni) di quelli montati dalla Sun sulle sue meravigliose workstation grafiche.
Apparentemente procurarsi questi pezzi sembrava facile, ma la credenza si rivelò presto errata. In particolare, nessuna ditta a mia conoscenza importava l'HardFrame 2000, e una breve ma approfondita ricerca mi confermò questa convinzione. Era giunto il momento di una scelta drastica: o lo ordino negli USA, o non se ne fa niente.
Forte di una certa esperienza nell'acquisto di software oltreoceano, mi misi a cercare la ditta che offriva la combinazione che cercavo a meno, e la trovai in una pubblicità su una rivista americana. Ma, nonostante la mia fiducia, le cose andarono in modo un pochino più contorto del previsto; e spero che la mia (tragica?) esperienza possa servire d'aiuto a chi seguirà i miei passi.
Prima di tutto mandai via fax l'ordine, ovviamente chiedendo l'addebito sulla mia carta di credito (non cercate di usare altri sistemi, perché sono sempre troppo rischiosi o troppo lenti; al limite, circuite un amico che possieda un fax e una carta di credito). Dopo circa un mese, nessuna notizia; cominciavo a preoccuparmi.
Un giorno ricevetti una gentile lettera (che naturalmente aveva impiegato parecchio tempo per arrivare), nella quale mi si chiedeva di mandare via fax una fotocopia del fronte e del retro della mia carta di credito. Quindi, lezione numero uno: dovunque e comunque ordiniate, mandate sempre insieme all'ordine il fronte/retro della carta di credito, e possibilmente, fate una fotocopia leggibile.
Finalmente, alla fine di Novembre avvenne il miracolo: un corriere mi portò a casa un magnifico pacco, caricato da 317.000 lire di tasse (IVA + dazio doganale). Lezione numero due: quando calcolate il prezzo, pensate sempre ad un 25% in più.
Era quindi giunto il momento magico di aprire il cartone, che conteneva, in effetti, un controller HardFrame 2000 e un hard disk Quantum ProDrive da 40Mb. La cosa strana (e inizialmente divertente) è che l'hard disk era contenuto in un sacchetto di plastica antistatico e basta. Niente istruzioni, niente informazioni, neanche un pezzetto di carta. Va bene, mi dico, ci sarà scritto tutto nel manuale della MicroBotics. Sì, ma quale manuale?
Colgo l'occasione per deplorare la pessima abitudine delle ditte che includono nel disco di setup dell'hardware un file, di solito con la beffarda estensione ".man", contenente il manuale del prodotto. Nel caso che l'acquirente (per esempio il sottoscritto) abbia appena esaurito il nastro della stampante, il montaggio dell'hard disk diventa un po' difficoltoso, in quanto non è semplice leggere le istruzioni da un computer spento.
Ma proseguiamo, dopo aver imparato a memoria (o quasi) il manuale mi accingo a montare il Quantum sulla piastra d'alluminio dell'HardFrame, in modo da risparmiare spazio, e, sempre seguendo il manuale della MicroBotics, cerco di attaccare con un cavetto in dotazione l'alimentazione dell'hard disk direttamente alla circuiteria del controller. Ma qui interviene il destino: la piastra d'alluminio che sorregge l'hard disk ha il bordo sagomato in modo tale da impedire fisicamente l'inserimento del connettore. Con che coraggio la MicroBotics (peraltro una ditta serissima) mette in giro un aggeggio del genere?
Dopo alcuni minuti di profonda disperazione, prendo in mano un taglierino da disegno e mi metto a sagomare il connettore tagliandone via qualche pezzo. In questo modo, riesco ad inserirlo al suo posto. Grezzo, ma efficace.
A questo punto pensavo che le mie

disavventure fossero finite. Non sia mai! Dopo aver inserito la scheda al suo posto, accendo il mio Amiga... e *non percepisco segni di vita*. L'hard disk si mette a girare con un sibilo sinistro, ma lo schermo resta bianco, i dischetti non cliccano e la manina non appare. Dopo un paio di reset, tutto ricomincia a funzionare, ma la mia angoscia cresce esponenzialmente.
A computer acceso diventa evidentemente un po' più semplice seguire il manuale. Preparo un paio di partizioni (ovviamente in Fast File System), le formatto e... urrà!, alla fine tutto sembra andare liscio. La velocità è strabiliante, sia in accesso che in trasferimento; preso da euforia cronica, comincio a copiare file su file, e dopo aver trasferito un po' di applicativi comincio i miei test: il DPaint viene caricato in circa 1.5 secondi (non male, eh?).
Sto già per raggiungere *il nirvana* informatico, quando, richiamando A-Talk III, appare il terrificante requester "Volume Programs has a read/write error": il sangue mi si *congela nelle vene*. Cancello il requester, e il programma viene caricato comunque, apparentemente in perfetta salute. Esco, e provo a caricare per più volte il DPaint. Dopo un paio di tentativi, di nuovo il tragico requester. E questa volta, il programma si pianta appena cerco di entrarci.
Non resta che dotarsi di calma e gesso, nonché ragionare con attenzione: cosa può causare il fenomeno? Dopo alcuni minuti, realizzo che il problema si riscontra solo con file molto grandi, tipicamente sopra i 200K. Volendo escludere ogni possibile causa estrinseca, smonto il 68010 dalla piastra madre e rimetto il 68000: niente da fare. Stesso risultato provando a modificare i buffer di cache del Fast File System. La *cosa veramente strana era che*, per quanto riguardava il resto, il disco funzionava a meraviglia.
Dopo due giorni di test non-stop, la risposta, troppo assurda per essere vera. Il Fast File System possiede un parametro di configurazione, detto MaxTransfer, che specifica il massimo numero di byte che deve cercare di trasferire in DMA in una sola passata, ovvero alla massima velocità disponibile; il programma di configurazione della MicroBotics lo metteva a 128K, e io, fiducioso, l'avevo lasciato tale e quale. Ciononostante, un MaxTrasfer di 130560 (=127.5K) fa funzionare perfettamente controller e hard disk, mentre uno di 131072 (= 128K) provoca errori di lettura. Qualche scambio di idee con la MicroBotics ha portato alla luce che una restrizione del genere sembra indotta dalla struttura a 16 bit dei registri hardware (65536 = 2^16 parole sono proprio 128K). Perché però tutto funzioni proprio a 127.5K non me l'ha saputo dire neanche il capo del reparto progetti, e quindi rimane un mistero. Se avete qualche idea, scrivete pure (mi hanno comunque detto che la versione attuale del programma di *configurazione usa il valore* 130560).
Naturalmente le mie tribolazioni non erano ancora finite: dopo alcune ore di prove, la piastra d'alluminio su cui era montato il drive scottava come un altoforno. Alcuni consulti telefonici (non sono un grande intenditore di hardware) mi condussero a sconnettere l'alimentazione diretta dalla piastra del controller, attaccando il Quantum direttamente all'alimentatore di Amiga. Il problema ora è quasi scomparso, sebbene ci sia sempre un certo riscaldamento.
Restava ancora il problema finale: all'accensione, ci volevano più di 30 secondi perché il tutto partisse, e, dopo un reset, almeno dieci. D'accordo, il disco rigido divorava ottanta linee di startup-sequence in nove secondi, ma perché dover aspettare di più?
L'intuito, ancora una volta, mi venne in aiuto: il programma diagnostico della MicroBotics diceva che il *Quantum era configurato come unità* SCSI numero sei (si possono collegare fino a sette dischi all'HardFrame). Era quindi probabile che il tempo perso fosse dovuto al fatto che il controller cercava di vedere se esistevano periferiche con numero 0, 1, 2, eccetera e solo arrivando alla numero 6 "vedesse" il disco rigido. Quindi, bisognava modificarne il numero di periferica, e, come in tutte le device SCSI, quest'azione doveva essere eseguita rimuovendo o aggiungendo dei jumper, cioè aprendo o chiudendo degli oppurtuni contatti. Fin qua, tutto bene: il problema era quello, dato che l'hard disk era giunto nudo e crudo. Potevo telefonare alla Quantum, ma erano le due di notte e prima di dormire il sonno dei giusti volevo riuscire a far andare tutto come si deve.
Detto, fatto: smonto nuovamente il disco dalla piastra del controller e cerco di indovinare quali potrebbero essere i jumper giusti; la fortuna mi aiuta, in quanto ce ne sono dappertutto variamente raggruppati, ma solo un gruppo è formato da tre elementi, e poiché il numero di periferica SCSI è definito esattamente *da tre cifre binarie*...
Credo di non aver avuto tanta paura di far fondere un computer come quando ho riacceso il mio fedele Amiga dopo avere estratto i due jumper presenti nel gruppo. Non credo però neanche di essere stato tanto felice come quando l'hard disk è partito a un paio di secondi dall'accensione e si è lanciato sulla startup-sequence.
Il bilancio definitivo, ovviamente, non è facile. Vorrei, innanzitutto, spendere qualche parola in favore del controller MicroBotics, che, a parte qualche problemino, se le merita.
HardFrame 2000 è perfettamente integrato nella filosofia multitask di Amiga; per esempio, viene gestito tramite una device standard (HardFrame.device) e quindi si inserisce elegantemente nella stuttura di Exec. Dispone di un buffer hardware interno, per non affollare inutilmente il bus di sistema, a cui *peraltro si attacca e stacca in tempi* brevissimi. La velocità grezza di trasferimento, come ho già detto, è esorbitante (40 megabit al secondo): unita al bassissimo tempo d'accesso del Quantum e al Fast File System dà risultati esaltanti. Per esempio, ho provato, contempora-

neamente, ad eseguire una copia da DH0:C a DH0:C2, ad aprire un cassetto con dentro circa 100 icone e a lanciare il DPaint: il rallentamento è stato quasi inavvertibile; gli accessi multipli che normalmente fanno gridare i floppy sono quasi "lenti", dal punto di vista dell'Hard-Frame. Oppure, il mio batch file "MakeCRes", che rende residente tutto il Lattice C più qualche utility, e carica quindi circa 560K, viene eseguito in 4.52 secondi. O ancora, leggendo con un normale programma C un file, la velocità di trasferimento dati è di circa 600K al secondo (QSpeed dà invece 731K/s).
Se poi in futuro vorrò prendere qualche altro disco rigido, posso collegarne fino a sette. Il software di gestione mi permette in qualunque momento di alterare i parametri delle partizioni, ad esempio la dimensione della memoria cache del Fast File System, e quest'ultimo è memorizzato insieme alle mountlist direttamente sull'hard disk, nella prima traccia: in questo modo non solo il drive esegue l'autoboot, ma anche l'automount, e scavalca il problema dato dal controller Commodore, che richiede per forza una partizione fatta con il vecchio File System per lavorare. Non ultimo, la MicroBotics è con Amiga da quando è nata, e quindi ha una vasta esperienza, al di là di alcuni scivoloni. Il vero problema è: quante persone sono disposte a sobbarcarsi le fatiche e a superare gli scogli che ho incontrato? E, ancora di più, quante potrebbero riuscirci? Quando la ditta da cui si è comprato un prodotto è sita a svariate migliaia di chilometri di distanza, non è così semplice tirare su il telefono e dire "Oops, non sono riuscito a montare niente...". E se qualcosa dovesse andare male (difetti di fabbricazione, danni durante il montaggio,...) l'avanti-indietro postale può durare parecchi mesi.
Nel complesso, e dopo alcune settimane di uso intensivo, sono molto contento della combinazione Hard-Frame/Quantum. Se siete disposti a soffrire e lottare per averla, vi consiglio di provare. In tutto viene a costare 1.500.000, che non è un prezzo eccessivo. Se però non siete esattamente dei patiti, forse è il caso di ripiegare su una più sicura (e soprattutto già montata) GVP, disponibile in molti negozi, sulla quale potete trovare anche montato un Quantum come il mio. Oppure aspettate che la Commodore metta in giro il 2091, che sembra essere vicino all'Hard-Frame come performance, e permette di montare 2Mb di RAM.

Pettegolezzi...

Mi resta ancora da narrare qualche pettegolezzo sparso. Prima di tutto, c'è giustizia a questo mondo: la Xerox ha fatto causa alla Apple per l'interfaccia utente del Macintosh, il che non era solo auspicabile, ma anche necessario per stroncare la tracotanza dei suddetti signori. Vedremo come andrà a finire.
Al posto di ARP, la 1.4 conterrà ASL.library, una libreria ricca di funzioni standard tra cui un File Requester. Inoltre, sarà disponibile una scalabilità limitata per i font bitmapped, in attesa dei font outline, che dovrebbero arrivare un po' dopo per terremotare il mondo del DTP e del word processing su Amiga. E su questo tono speranzoso vi do appuntamento alla prossima volta...

Ma...

Proprio mentre finivo di scrivere questo articolo, ho scoperto che una ditta, la Creative Computers, non solo vende software e hardware per Amiga a prezzi ottimi, ma permette anche di eseguire ordini, se superiori a 100$, attraverso un numero verde (1678-74086): questo significa che potete fare un'ordinazione per telefono mediante carta di credito spendendo un solo gettone. Badate solo a calcolare il fuso orario e tenete pronti i vostri dati. Sono piuttosto gentili, ma non cercate di chiedere informazioni al telefono. ▲

(segue da pag.15)

TOTOAMIGA: Come vincere al Totocalcio, Enalotto e Totip

Anche altre più sofisticate, quali le interruzioni particolari sul pronostico, le sequenze di segni a passo variabile, l'analisi del sistema in base a formule derivate fanno grande questo prodotto. Il tutto si gestisce con l'uso del solo mouse tanto da rendere quasi inutile l'impiego della tastiera. Le varie opzioni sono altresì richiamabili in ogni momento e si presentano con l'aspetto grafico dell'immagine sull'immagine.
Il numero delle colonne ottenibili è assai elevato (arriva a 60000) e tutte sono visualizzabili con la stampa grafica a monitor. Le condizioni che vogliamo stabilire per un determinato sistema possono essere memorizzate per un uso futuro.
Un punto negativo è stato riscontrato nell'uso della funzione di compattamento che raggruppa più colonne del sistema memorizzato in sistemi più piccoli; è lodevole la sua presenza ma, purtroppo, richiede tempi di elaborazione estremamente lunghi. Aggiungiamo, infine, che l'uso del programma è esteso anche allo sviluppo di sistemi per l'enalotto e per il totip.
In conclusione possiamo essere certi che il rapporto prestazioni-prezzo di un tale programma risulta estremamente vantaggioso: i pregi sono la chiarezza d'uso e la completezza delle funzioni. Rimane tuttavia sempre la perplessità che talvolta affiora sulla affidabilità di Amiga nell'uso di programmi particolarmente sofisticati; affidabilità che in macchine come i PC compatibili è quasi sempre assoluta.

TOTOAMIGA è in vendita in tutte le edicole al prezzo di L. 15.000.

Citizen Swift 24

Stefano Paganini

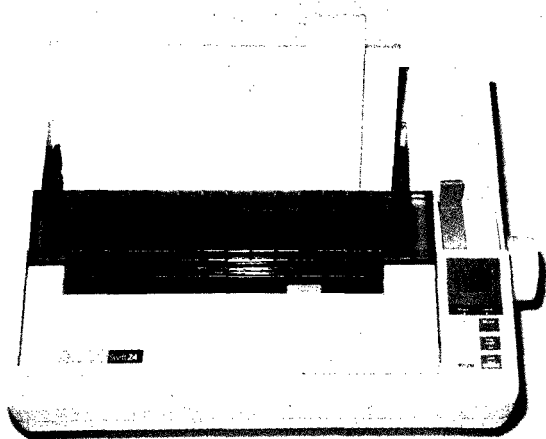
Sono disponibili sul mercato, ormai da diversi anni, stampanti a colori con una buona gamma di prestazioni e prezzi al pubblico. Tuttavia vi sono ancora parecchi problemi che, molto spesso, impediscono all'utente il pieno utilizzo delle potenzialità della loro stampante, in particolare per quanto riguarda le stampe con una certa gamma cromatica. Per gli utenti di Amiga il problema si pone in termini ancor più evidenti: si tratta, infatti, di rendere l'hard copy di immagini che a schermo hanno una gamma di colori compresa tra i 16 delle modalità ad alta risoluzione fino ai 4096, con le ben note complicazioni, delle modalità hold & modify.

La Swift 24

La stampante in prova questo mese, la Citizen Swift 24, si pone quale valida soluzione a questo e ad altri problemi di stampa. La Citizen Swift 24 è una stampante da 80 colonne a 24 aghi, un massimo di 192 cps e dispone di 5 font Letter Quality residenti in memoria: Roman, Sans Serif, Courier, Prestige e un font draft. E' previsto l'utilizzo in emulazione di 3 stampanti: Epson LQ, IBM Proprinter e Nec P6 +; in quest'ultima modalità, è, inoltre, possibile settare la densità fino a 360 dpi.
Tra le opzioni vanno citate la scheda per l'interfacciamento RS232, oltre alla parallela Centronics fornita di serie; le font card opzionali contenenti ulteriori corpi e la possibilità di espandere il buffer di stampa dagli 8 Kb di base fino a 40 Kb oltre, naturalmente al kit colore.
Una delle peculiarità più interessanti della Swift 24, consiste però nella possibilità di modificare tutti i parametri di stampa tramite un sistema di menu e sottomenu visualizzati sul display a cristalli liquidi presente in prossimità dei sei tasti funzione a membrana. I menu disponibili sono ampiamente documentati sui manuali in dotazione, un Reference e una Quick Guide, dei quali solo il secondo ha anche una parte in italiano. Inoltre, i menu sono abbastanza intuitivi e, dopo qualche prova non presentano più problemi, anche perché la maggior parte dei setting sono effettuati "una tantum", cioè solo all'atto dell'installazione. E' possibile, in ogni momento, cambiare la configurazione di parametri e funzioni quali, ad esempio: font, densità, stile e colore del testo, qualora sia inserito il kit colore, la presenza del quale è "avvertita" automaticamente dalla stampante stessa. Vi sono, poi, dei parametri riguardanti il layout e le dimensioni del foglio; a tale proposito, è da notare che la Swift 24 accetta sia moduli continui che fogli singoli A4, con la possibilità di parcheggio del foglio. Sono previste, inoltre, quattro Macro definibili dall'utente, che non sono altro che configurazioni di uso frequente richiamabili in blocco con la pressione di un tasto, e che risultano comode qualora utilizzate con programmi differenti e con esigenze di stampa.

La prova su Amiga

Veniamo ora alla "prova su strada" vera e propria con Amiga: i problemi principali riguardavano i driver: la scelta è caduta principalmente su EpsonQ tramite Preferences 1.3, modificando gli altri setting quali l'opzione colore e, di volta in volta, il dithering e la correzione del colore, anche se i parametri di default si sono rivelati più che soddisfacenti.

Hard Copy di un'immagine in Hold & modify

Per la prova abbiamo fatto uso dei programmi grafici di più ampio utilizzo tra quelli per Amiga, ottenendo sempre hard copy di buona qualità. Qualche problema è sorto con immagini in hold & modify che richiedono qualche accorgimento, non solo a livello di Preferences, come già detto, ma anche a partire dall'immagine stessa, dalla quale devono essere eliminate quante più discontinuità cromatiche sia possibile. A tale scopo sono molto utili sia Deluxe Photolab sia Pixmate; il primo, in particolare, permette, tramite il programma Poster, di stampare qualsiasi file IFF su superfici superiori o inferiori al foglio A4 con estrema precisione, mentre il Pixmate contiene probabilmente le opzioni più avanzate per l'image processing in HAM..
Una tra le caratteristiche più apprezzate del Deluxe Photolab Poster, è la possibilità di selezionare il bianco come background, e una volta effettuata la campitura degli eventuali bordi con un registro colore bianco, con ciascuno dei tre valori RGB al massimo, si può economizzare in maniera non indifferente sul costoso nastro... Vi sono poi dei programmi particolari, quali ad esempio IntroCAD, che fanno uso di driver propri, ad esempio GenericQ, ed è in questi casi che può tornare utile la multi-emulazione, tanto più che come NEC P6 si ottiene la massima precisione di stampa, soprattutto in B/N. Nessun problema nemmeno con Photon Paint 2, anche se qualche piccola imperfezione può già risaltare con 64 colori.
A questo punto pare evidente che la scelta cromatica a video deve essere quantomai accorta, tanto più che non è possibile intervenire direttamente e modificare sostanzialmente il contrasto e la chiarezza dell'immagine stampata; anche in questo caso la pratica si rivela insostituibile, suggerendo per ogni immagine una soluzione ad hoc. La qualità di stampa è decisamente buona nella maggior parte dei casi e si allinea coerentemente con le altre prestazioni della stampante al suo costo al pubblico che, incluso il kit colore, è di circa 1.250.000 lire, collocandola in una fascia superiore tra le stampanti ad impatto.

Una stampante molto compatta ma dalle elevate prestazioni

Conclusioni

In conclusione, si tratta di una stampante dedicata ad un'utenza molto ampia, dove sono inclusi anche utenti con precise esigenze di output, che trovano nella Swift 24 un'ottima macchina da abbinare ad Amiga, sia nelle consuete stampe che nella modalità a colori la quale, pur conservando le ovvie limitazioni della tecnologia ad impatto, riproduce con accuratezza i risultati delle applicazioni di Amiga. ▲

Citizen Swift 24 è in vendita presso: Digimail - Tel. 02/426559

Music-X: una potente WorkStation MIDI per Amiga
(Prima parte)

Aldo & Andrea Laus

Negli USA, dove è stato sviluppato dalla MICROILLUSIONS, è già in circolazione da più di un anno. In Inghilterra è arrivato a metà dell'89 e, da qualche mese, finalmente, con il supporto della nota società distributrice Soundware S.r.l., è disponibile anche sul nostro mercato. Le presentazioni ufficiali sono avvenute, in grande stile e con un grosso successo, all'ultima edizione dello SMAU, con tanto di gruppo musicale in azione presso lo stesso stand della Commodore. Descrivere in dettaglio le capacità e le prestazioni di questo pacchetto, nello spazio limitato di un articolo, non è cosa facile, in quanto MUSIC-X non è uno dei soliti programmi sequencer, anche se quello del sequencer è il suo ruolo principale, ma si presenta come una completa raccolta di programmi e sotto programmi dedicati alla gestione di una moltitudine di funzioni MIDI che, nella nostra esperienza, mai, prima d'ora, abbiamo incontrato, sia su Amiga che su altri computer. Cercheremo quindi di darvi una descrizione essenziale delle principali caratteristiche e delle applicazioni che noi intravvediamo facilitate dalle sue particolari prestazioni. Focalizzeremo poi, sempre in questo articolo, il suo uso come sequencer, rimandando in seguito la recensione delle altre funzioni. Il pacchetto MUSIC-X consiste di 3 dischetti contenenti rispettivamente: il programma principale, alcune utility e degli esempi applicativi. E' adatto ad ogni Amiga (500, 1000, 2000), una volta dotato di interfaccia MIDI. Il manuale, di circa 220 pagine, è tradotto in italiano e, oltre a spiegare l'utilizzo del software, si propone come un'utile guida al mondo MIDI e relative applicazioni, per chi inizia a muoversi in questo campo. Tali tutorial sono distribuiti all'interno del manuale in corrispondenza delle descrizioni delle varie funzioni e prestazioni del programma. Approviamo e sosteniamo questo approccio che, sicuramente, aiuterà i nuovi utenti, non solo nell'utilizzo, ma anche e soprattutto, nella comprensione dei vari argomenti. MUSIC-X si presenta come un programma costituito, per ora, da 4 moduli principali (definiti pagine) e da 4 submoduli interconnessi fra di loro (vedi Fig. 1), con la possibilità di ampliamenti futuri, per mezzo di nuovi moduli. Sul manuale, infatti, rileviamo la promessa di nuovi moduli, dedicati ad applicazioni audio/video.

Architettura del programma

Nel numero 4 di Amiga Magazine, abbiamo descritto le varie tipologie di programmi MIDI e il relativo impiego da parte del musicista (sequencer, editor, librarian, ecc.). Salvo rari casi, finora, un musicista doveva dotarsi di diversi di questi programmi per operare nel suo studio MIDI. Il principio ispiratore di MUSIC-X è sicuramente stato quello di realizzare un potente tool che, grazie a numerose applicazioni, potesse soddisfare tutte, o quasi, le esigenze del musicista MIDI, senza che questi si trovi costretto a ricorrere ad altri software dedicati. Vediamo ora, più in dettaglio i vari blocchi del MUSIC-X, con riferimento alla Fig. 1 (rimandiamo al nostro articolo sopracitato per le spiegazioni di base dei vari tipi di programmi).

1) PAGINA SEQUENCER, con possibilità di EDIT degli eventi, sia in modo grafico (BAR EDITOR) che in forma di listato degli eventi (EVENT EDITOR). Di seguito descriveremo in dettaglio le prestazioni del sequencer.

2) PAGINA FILTRI che consente di

selezionare i messaggi MIDI in input e indirizzarli a piacere, secondo le esigenze. Da questa pagina si ha accesso al submodulo KEYMAP EDITOR che permette di assegnare funzioni e comandi MIDI alla tastiera dello strumento musicale collegato oppure assegnare alla stessa il pilotaggio del sequencer. Questa funzione, da sola, vale, secondo noi, l'intero costo del programma in quanto, a prescindere dal suo uso con il sequencer, può rappresentare un valido supporto in tempo reale per qualsiasi studio MIDI.

Figura 1

3) PAGINA AMIGA SAMPLE, dedicata alla gestione dei suoni interni di Amiga. Qui potete caricare suoni, sia in formato IFF che SONIX e utilizzarli (li potete anche modificare), pilotandoli da una o più tracce del sequencer. Ciò significa che MUSIC-X può far suonare o sorgenti MIDI o i generatori interni di Amiga o entrambi. Questo modulo giustifica l'acquisto di MUSIC-X da parte di chi non ha ancora comprato strumenti MIDI, poiché rende utilizzabile il sequencer anche solo con le voci interne. In ogni caso, pur possedendo già strumenti MIDI, se non avete un campionatore, vi consente di usare Amiga come tale, mentre gira il sequencer.

4) PAGINA LIBRARIAN, consente di caricare dai sintetizzatori (e successivamente di scaricare) i dati relativi alle singole voci (programmi), tramite il sistema esclusivo MIDI, e di salvarli su disco. In questo modo è possibile realizzare un vero e proprio archivio di voci, richiamabile direttamente da MUSIC-X. Siccome però, ogni strumento (ahimè) ha un protocollo diverso, è prevista, nel dischetto Esempi, una dotazione di protocolli relativa agli strumenti più diffusi. Per protocolli diversi, esiste una funzione PROTOCOL EDITOR con la quale, facendo riferimento al manuale del proprio strumento, è possibile risolvere il problema. La Soundware è comunque a disposizione per aiutare, con ulteriore documentazione, gli utenti più evoluti che vogliano cimentarsi da soli per allargare tale raccolta. Il sottomodulo INDIVIDUAL PATCH EDITOR, consente la modifica dei dati dei suoni immagazzinati dal Librarian per creare nuove voci o per modificare quelle già esistenti e, per alcuni strumenti, presenta uno schema grafico su cui agire direttamente. Se l'utente possiede strumenti diversi, deve ricorrere al modulo GENERIC PATCH EDITOR, che gli fornirà il listato dei dati relativi al suono su cui lavorare (auspichiamo, a tale proposito, che siate ferrati conoscitori dell'esadecimale). Ulteriori INDIVIDUAL PATCH EDITOR dovrebbero essere in sviluppo con l'avvento di nuovi strumenti sul mercato. Passiamo ora ad esaminare la funzione principale di MUSIC-X, ovvero le pagine SEQUENCER ed EDITOR.

1

Il Sequencer

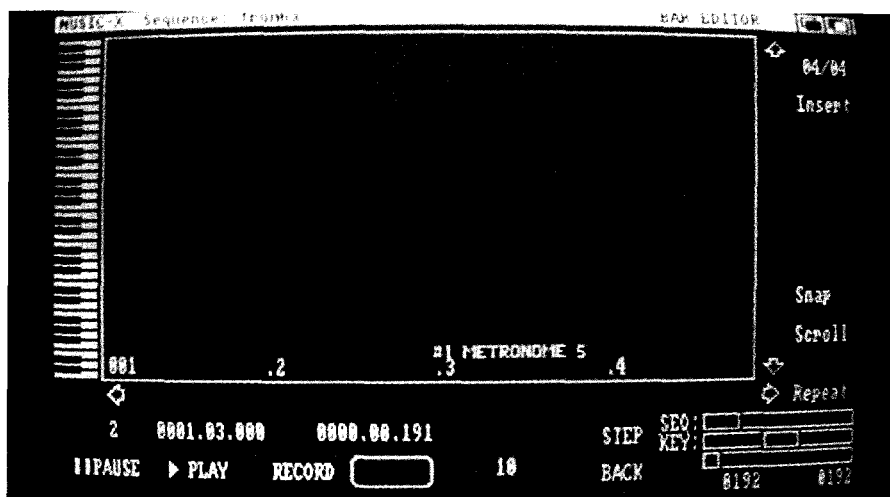
Senza dubbio, la pagina Sequencer è la più importante del MUSIC-X e, comunque, quella da cui si comincia a lavorare (Foto 1).
E' costituita da un certo numero di finestre che contengono i comandi e le indicazioni essenziali. Vediamole in dettaglio:

La BARRA DEL TITOLO vi indica, nella parte sinistra, il nome della performance (brano) corrente, mentre sulla parte destra è indicata la pagina del programma in cui vi trovate (nel nostro caso indica "Sequencer"). Cliccando il pulsante destro del mouse, la barra vi presenta i menu a cui potete accedere. I veri e propri comandi del sequencer hanno la classica impostazione di quelli del registratore a nastro e sono contenuti nella "FINESTRA DEL TRASPORTO" (parte alta a sinistra dello schermo). Da qui è possibile iniziare la registrazione in tempo reale (lo step by step si attiva, invece, dalla pagina EDIT). Molto utili i comandi CUE 1,2,3,4 che, definibili dall'utente, posizionano la song velocemente, avanti o indietro, in punti prestabiliti. Immediatamente sotto c'è la "FINESTRA DEL CLOCK" che contiene due visualizzatori digitali. Il primo, chiamato CLOCK è quello del tempo Relativo (musicale) e visualizza misure, battiti e clock (1 clock=1/192 di 1 nota da 1/4). Il secondo, chiamato TIME è quello del tempo Assoluto ed è espresso in ore, minuti, secondi e frame. Sotto alla finestra precedente c'è la "FINESTRA DEL TEMPO" che contiene queste funzioni:

- Cursore del tempo: visualizza la regolazione in note di 1/4 al minuto in un range da 1 a 300 (può essere regolato in tempo reale, sia durante il Play che il Record).

- Indicatore di memoria: dà la disponibilità di memoria in numero di eventi MIDI registrabili. La "FINESTRA TIME SIGNATURE", a destra della finestra del tempo indica la divisione del tempo corrente ed è modificabile in un range abbastanza ampio di tempi. La metà inferiore della Pagina Sequencer viene utilizzata dalla finestra che contiene la lista delle sequenze. Tale lista ospita fino a 250 sequenze, che è la capacità massima del MUSIC-X. Vediamo brevemente, a questo punto, come è organizzata la registrazione nel MUSIC-X. L'idea basilare di un sequencer MIDI è quella di mettere un compositore in condizione di poter registrare parti strumentali una alla volta, modificarle e suonarle tutte tramite strumenti MIDI. Queste procedure sono analoghe a quelle che vengono usate operando con un registratore multitraccia a nastro. Ci sono vari modi per assemblare un brano musicale con un sequencer. Alcuni preferiscono registrare ogni strumento in una sequenza (dalla prima in poi) lunga quanto tutto il brano (Song, usando il termine dei sequencer MIDI). Altri preferiscono sovrapporre gli strumenti in una singola sequenza lunga quanto una sezione del brano e, successivamente, assemblare tutte le sezioni per creare il brano. Fortunatamente, il MUSIC-X è stato scritto per soddisfare tutte e due le esigenze. Una sequenza può essere lunga quanto tutto il brano (fino a 4096 battute), o quanto una sola battuta; può, inoltre, contenere dati per uno strumento (o, meglio, per un canale MIDI) o per più di uno di essi; può anche contenere eventi che possono attivare altre sequenze. Sono disponibili 250 sequenze e ne possono essere suonate contemporaneamente fino a 20. Combinando tutte queste caratteristiche, può essere emulata qualsiasi modalità relativa alla composizione della musica tramite un sequencer. Alcune delle varie modalità vengono descritte nei dettagli nel capitolo "Utenti Avanzati" del manuale.
La sequenza selezionata, col mouse o con i tasti cursore, è evidenziata e diventa la "sequenza corrente", su cui agiscono tutte le operazioni o i menu che hanno effetto sulle sequenze. Ogni sequenza è identificata da una serie di dati informativi, distribuiti nei campi della finestra. Essi, in sintesi, indicano:

SEQ
Numero progressivo della sequenza (1-250)
MEM
Numero di eventi occupati
BARS
Numero di battute occupate
CHANNELS
Numero di canali MIDI (anche più di 1 per sequenza)
CTS
Presenza di eventi MUSIC-X per il controllo interno del sequencer.
TIME
Formato del tempo prescelto per registrare le sequenze (REL = relativo ; ABS = assoluto)
NAME
Contiene il nome che l'utente assegna ad ogni sequenza (max 27 caratteri)
OUT
Indica su quale uscita volete inviare i dati. E' possibile selezionare INT per scegliere le voci di Amiga, oppure EX per uscire sul bus MIDI. Qui è prevista la possibilità futura di selezionare altri tre bus MIDI di 16 canali ciascuno.
FRECCE
Questo campo consente due ope-

razioni: la prima OFFSET disattiva una sequenza in modo da non farla suonare (è possibile, però, attivarla inviando un comando PLAY SEQUENCE da un'altra sequenza). La seconda ritarda o anticipa l'attivazione di una sequenza di un numero di battute da specificare (molto utile per risparmiare memoria)

L'ultima finestra della pagina Sequencer è la "FINESTRA DELLE TRACCE", che si trova nella parte superiore destra dello schermo. Questa è utilizzata per visualizzare lo stato di ciascuna delle 20 tracce del MUSIC-X. Qui conviene chiarire che cos'è il concetto di traccia per il MUSIC-X, dato che, visto il grande numero di programmi sequencer sul mercato, bisogna capire bene quale funzione operativa e significato ciascuno di essi attribuisce ai vari termini. Per il MUSIC-X una traccia è quell'insieme di informazioni, costituito da una o più sequenze che verrà suonato al momento della pressione del tasto Play. Facciamo un esempio per chiarire meglio il concetto. Una sequenza può essere considerata come un pezzo di nastro, mentre la traccia come la testina di playback che ci permette di ascoltare il contenuto della sequenza. Solo un pezzo di nastro può essere suonato da una testina di playback alla volta, ma, durante quest'operazione, passano differenti pezzi di nastro. Una traccia può solo suonare una sequenza alla volta, ma non è dedicata ad una sola sequenza. Considerate che una batteria elettronica può suonare molte sequenze brevi una dopo l'altra, in modo da creare una song lunga. Il MUSIC-X è un sequencer a 20 tracce: può suonare contemporaneamente fino a 20 sequenze. Se pensate che questa possa essere una limitazione, ricredetevi: non lo è. Infatti, è possibile miscelare una o più sequenze in una sola, senza per questo che le singole sequenze perdano la loro individualità e le loro caratteristiche (tra l'altro è possibile estrarre una sequenza, una volta miscelata). In questo modo aumenta il numero, non di sequenze, ma di

parti musicali suonate contemporaneamente. Se ancora non vi basta (ma cosa dovete suonare ?), è virtualmente possibile far suonare una sola sequenza alla volta che contenga, anziché dati musicali, informazioni e comandi per far suonare tutte le sequenze che volete. Vi basta? La finestra delle tracce è caratterizzata da due campi: BAR e SEQ. Il primo indica (in Play) la battuta che viene suonata in quel momento, il secondo il nome della sequenza che sta suonando. Entrambi scompaiono quando nella sequenza non è più inciso niente o quando, tramite un'altra sequenza, viene disattivata quella sequenza. A questo punto possiamo passare ad esaminare gli Editor.

Le pagine Editor

Un Editor permette di esaminare, visualizzandolo, il contenuto di una sequenza registrata in precedenza e di manipolarla a piacere; oppure è possibile crearne una nuova (step by step). Una sequenza può contenere eventi quali messaggi di nota on/off, program change, comandi vari, ecc.. Il MUSIC-X è dotato di due editor per le sequenze: il BAR EDITOR e l'EVENT EDITOR; lo scopo è quello di rappresentare gli stessi eventi in due modi diversi. E' consentito spostarsi fra le due pagine liberamente, utilizzando e sfruttandole diverse capacità di ciascuna di esse. L'editor lavora su un buffer di memoria a parte; significa che potete fare tutti i vostri esperimenti prima di decidere se adottare o meno le modifiche apportate alla sequenza originale. Il BAR EDITOR visualizza gli eventi graficamente, rappresentandoli come icone posizionate in un grafico bidimensionale i cui parametri sono: il tempo in orizzontale e, ad esempio, le note in verticale. Le note compaiono come barrette colorate. Per aggiungere, rimuovere o spostare gli eventi sul grafico si usa il mouse a guisa di pennello, come nei programmi grafici. Il colore delle note dipende dal canale MIDI a cui esse sono indirizzate. Altri eventi che possono essere rappresentati a scelta sono:

ATTACK VELOCITY, sotto forma di rettangoli, posizionati sotto la nota in questione, la cui altezza esprime, proporzionalmente, il valore delle velocity.

RELEASE VELOCITY: come il precedente ma rappresentabile in alternativa ad esso.

PROGRAM CHANGE: compare il numero dei vari programmi preceduto da "PGM" nel punto appropriato del diagramma della sequenza; il colore è in funzione del canale.

CHANNEL AFTER TOUCH: compaiono dei punti, ad altezze proporzionali alla intensità, collegati alla base del diagramma da linee verticali (simili a spilli); il colore è in funzione del canale.

POLIPHONIC AFTER TOUCH: il simbolo è analogo al precedente e vie-

ne mostrato in alternativa ad esso.
CONTROL CHANGE & PITCH BEND: vengono visualizzati i relativi eventi.
SYSTEM EXCLUSIVE: vengono evidenziati i dati sotto forma di piccoli trattini verticali di colore nero. E' possibile, e consigliabile, visualizzare individualmente i vari eventi della sequenza, altrimenti, data l'alta densità di informazioni sul video, si rischia di non riuscire a capire cosa succede, anche se con la funzione zoom è possibile discriminare meglio la schermata. I comandi, per tutte le funzioni, si selezionano sia a menu che sullo schermo. L'EVENT EDITOR, invece, rappresenta sullo schermo un listato contenente l'elenco di tutti gli eventi della sequenza in ordine cronologico. Ogni evento viene visualizzato e i parametri relativi sono suddivisi in vari campi. Si può far scorrere l'elenco per esaminare la successione e modificare, aggiungere o togliere gli eventi. In altre parole, il suo funzionamento è simile a quello di un word processor. La pagina dell'EVENT EDITOR è divisa in tre finestre. Quella più grande contiene l'elenco degli eventi. A destra c'è la finestra che mostra i comandi disponibili. Altra finestra mostra i tipi di eventi e i canali MIDI utilizzabili durante la fase di editing. In fondo alla pagina ci sono i controlli del trasporto. Si può lavorare sia con il mouse che con la tastiera. Dalle pagine Editor è possibile eseguire una miriade di operazioni che non possiamo descrivere in dettaglio in questa sede, ma che, oltre a tutte quelle tipiche dei vari sequencer (CUT, COPY, PASTE, PUNCH IN/OUT, TEMPO CHANGE, ecc..), ne comprendono anche di nuove e originali che utilizzano le possibilità offerte dall'ambiente di Amiga.

Conclusioni

Per questa volta ci fermiamo qui nella descrizione delle funzioni del programma MUSIC-X, con la promessa, però, di continuare al più presto. Prima di chiudere vogliamo comunque fare qualche considerazione, dopo aver utilizzato, sia pure per poco tempo, il MUSIC-X. MUSIC-X è stato progettato per Amiga sin dall'inizio e ciò è dimostrato dalle molteplici possibilità offerte dal Multitasking. La possibilità quindi di saltare da una pagina all'altra e di coprire tutte le esigenze con un solo pacchetto software è unica! La grafica è ben curata e il funzionamento, nel corso delle prove, si è sempre dimostrato sicuro. Un fattore molto importante nella scelta di un programma risiede anche nella valutazione del supporto che lo sostiene dal punto di vista dell'assistenza. Ciò vale in particolare quando l'oggetto si presenta come un prodotto aperto, suscettibile quindi di ampliamenti, come il MUSIC-X, per il quale la stessa MicroIllusions si impegna a realizzare gli aggiornamenti e le espansioni man mano che si evolve la tecnologia musicale. Sembra, fra l'altro, già in cantiere un modulo score writer.In questo caso dobbiamo dire che ci è piaciuto quanto Soundware ha già fatto e quanto dichiara di fare nel manuale, che possiamo così sintetizzare:

- Manuale ben curato in italiano
- Hot Line telefonica per consulenze rapide (0332/222052)
- Cartolina per registrazione utente/garanzia per futuri aggiornamenti e iniziative della Soundware
- Aggiornamenti per protocolli di sistemi esclusivi di nuovi strumenti
- Disponibilità di informazioni aggiuntive per "utenti evoluti"
- Sviluppo in corso per applicazioni che prevedono l'interazione tra il MUSIC-X e il mondo video. MUSIC-X, come ogni pacchetto software complesso, non si impara ad usarlo leggendo solo il manuale, né si può pretendere di utilizzarlo senza studiarsi le istruzioni.

La soluzione che suggeriamo è di fare pratica iniziando con un approccio soft, un po' per giorno, con la moltitudine di opzioni e possibilità operative offerte. E' un programma che può essere usato in molti modi e per scopi diversi e non necessariamente come sequencer. Ogni esempio, a questo punto, sarebbe riduttivo, rispetto alle reali possibilità, ma vogliamo comunque citarne qualcuno: a volte, nei setup MIDI, è utile disporre, per suonare in tempo reale, della funzione KEY MAPPER. La prestazione forse più banale, resa possibile da tale funzione, è quella dello split della tastiera in varie sezioni, ciascuna assegnata a canali MIDI diversi. Ciò si rivela assai utile per pilotare diversi expander o diverse voci dello stesso expander, nel caso in cui disponiate di una tastiera con implementazione MIDI non molto evoluta, che trasmetta, per intenderci, su un solo canale. Potrete ottenere tutto ciò nel modulo KEYMAP EDITOR. Un altro esempio è quello di assegnare agli ultimi 16 tasti a destra della vostra tastiera (o master keyboard) il comando di nota di 16 voci di una drum machine, associando quindi a ciascun tasto la nota e il canale MIDI dell'effetto voluto. E se non possedete una drum machine? Non c'è nessun problema: procuratevi dei sample IFF (o anche SONIX) per Amiga e assegnate i tasti della vostra tastiera alle voci interne. Altra possibilità, assegnare a qualunque tasto un qualunque comando MIDI. Ad esempio: Start per la batteria, Sustain, Numeri di program change e perfino comandi per pilotare da tastiera lo stesso sequencer. Tutte queste funzioni, facenti parte del KEYMAP EDITOR, submodulo della pagina FILTRI, vi consentono di risparmiare i soldi dell'acquisto di un costoso Mapper hardware. Non ci resta spazio e non avrebbe senso ora citare solo qualcuna delle innumerevoli prestazioni offerte dal programma.

Ritorneremo magari in futuro con articoli ad hoc su ciascuna delle pagine del MUSIC-X, per esaminare, con esempi pratici, cosa si può ottenere. Riteniamo però che, quanto esposto, anche se in maniera succinta, riesca già a dare un'idea di cosa sia ancora possibile fare con questo straordinario pacchetto.
In ogni caso, per ulteriori informazioni o per approfondimenti o chiarimenti urgenti, vi consigliamo di approfittare della HOT-LINE SOUNDWARE, telefonando al numero: 0332/222052. ▲

ARP 1.3

Cominciamo da questo mese una nuova rubrica che tratterà i programmi di pubblico dominio.

Sebastiano Vigna

Per chi non lo sapesse, sotto questa etichetta (o sotto quella inglese equivalente di Public Domain, PD) sta tutto il software che l'autore distribuisce liberamente senza richiedere alcun compenso. Contrariamente a quanto si potrebbe pensare, non si tratta sempre di programmi di qualità scadente; anzi, le utility più interessanti per Amiga sono quasi tutte nel PD.
In particolare, questa volta parleremo di uno dei più importanti pacchetti esistenti sotto Amiga, vale a dire ARP. Non è proprio esatto dire che ARP è PD, in quanto gli autori mantengono il copyright, ma in pratica tutti i suoi file sono liberamente distribuibili (si possono facilmente recuperare attraverso i BBS italiani).
AmigaDOS (o AmigaDOG, come alcune malelingue dicono) è notoriamente il peggior DOS esistente sul pianeta Terra. E' stato scritto separatamente dal sistema operativo di Amiga, Exec, con un linguaggio diverso e criteri diversi (per la precisione, peggiori). Poiché la lettura/scrittura dei dischi è evidentemente una parte essenziale di qualunque pacchetto applicativo, la cosa rappresenta una palla al piede non da poco, e non si capisce perché la Commodore abbia aspettato la 1.4, tra l'altro non ancora in vista, per riscriverlo. Nel frattempo, però, la comunità Amiga non ha dormito.
In particolare, un gruppo di programmatori, capeggiato da Charlie Heath, ben noto per i Fast Fonts e per l'editor TxEd, ha pensato di riscrivere AmigaDOS. Da qui il nome ARP, acronimo di AmigaDOS Resource Project (da questa release "Resource" ha rimpiazzato il precedente "Replacement"; con questo, Charlie Heath ha voluto indicare che ARP si pone in un rapporto di collaborazione con la Commodore).
Il progetto comprende due parti: la prima è una libreria estremamente compatta (17K) scritta in un Assembler ottimizzato che duplica tutte le funzioni di AmigaDOS e ne aggiunge letteralmente a dozzine. Tra l'altro, mette a disposizione una completa gestione asincrona dei processi, cosa fondamentale in una macchina multitask (per chi non lo sapesse, è pressoché impossibile per un programma sotto AmigaDOS lanciarne un altro e poi tornare a seguire il proprio corso con un minimo di sicurezza). Proprio sulla potenza di questa libreria si basa la seconda parte del progetto, vale a dire un rimpiazzo per tutti i comandi AmigaDOS, una decina di comandi in più e un'eccellente shell. I comandi sono completamente compatibili con la versione 1.3 del sistema operativo, ma dispongono in generale di potenzialità aggiuntive; inoltre, si tratta sempre di programmi più brevi e più veloci. Un esempio: il List di AmigaDOS è lungo 9972 byte, mentre quello di ARP 2500. Solo che quest'ultimo permette di ordinare alfabeticamente l'uscita, di usare molti più wildcard (come il "*" di Unix e MS-DOS) e dispone di un help in linea più sofisticato.
L'occasione per parlare a fondo di ARP viene dal fatto che negli ultimi mesi ha visto la luce la release 1.3, che rappresenta un nuovo fiore all'occhiello per il gruppo. Purtroppo, non ovunque il verbo di ARP si è sparso: molti non sanno nemmeno cosa sia, e sono costretti a sbattere la testa contro il muro quando devono rinominare trenta file da

<qualcosa>.txt a <qualcos'altro> .bak. Oppure quando devono togliere la flag di protezione a un centinaio di file (sì, lo so, si può usare PAT, ma provate a contare quanto ci mette). Oppure quando devono aggiornare qualche dischetto con le nuove release del sistema operativo o di un applicativo (sotto ARP "Copy df1:c df0:c opt o" copierà nella directory df0:c tutti i file in df1:c che già esistono in df0:c). Si potrebbe continuare per pagine intere. Non tratterò in esteso le funzioni della libreria, in quanto costituiscono un argomento a sé stante, e meritano un discorso approfondito. Cito però la funzione FileRequest(), che permette di attivare un bellissimo file requester, da molti (e da me) considerato il migliore disponibile su Amiga. Il requester è utilizzato, fra l'altro, dal TxEd e dal disk editor DiskX di Steve Tibbet. L'idea che consente un salto di qualità rispetto alla media è che premendo il tasto destro del mouse o il gadget "Drives", vengono mostrate tutte le periferiche logiche e fisiche del sistema, il che consente un accesso lampo a qualunque directory. Non mancano ovviamente una scroll-bar in tempo reale e un paio di gadget a freccia con autorepeat. In molti programmi avviene qualcosa di simile, ma vengono viste solo le periferiche fisiche e, in generale, il posto disponibile è prefissato; tanto per fare un esempio, DPaint III mi mostra sì RAM:, RAD:, DF0:, DF1: e PATH:, ma si dimentica (per mancanza di spazio) di DH0: e DH1:, il che, ovviamente, è piuttosto irritante. Una menzione merita anche l'inclusione di alcune classiche funzioni C, come Printf, Puts e Strcat, che permettono di non dover includere in ogni programma il codice di libreria. In effetti, la mia esperienza personale dice che scrivere comandi di sistema sotto ARP porta automaticamente ad una riduzione del 50% e più delle dimensioni del programma. Ad esempio, il parsing della linea di comando viene eseguito in modo completamente automatico da ARP tramite la funzione GADS (Get Arguments in Dos Style): una volta specificato il template AmigaDOS (ad esempio, "Name,Dir,Etc/..., LIST/s,EXISTS/s,REMOVE/s" per Assign), gli argomenti vengono automaticamente inseriti al loro posto, indipendentemente dall'ordine con cui sono stati scritti dall'utente.

I comandi ARP

La prima cosa da mettere in evidenza è che il fatto di disporre di una libreria di base così potente innalza immediatamente la qualità di tutto il software che la sfrutta. Un primo esempio sono i wildcard, come già accennato: oltre alla totale compatibilità con AmigaDOS, ARP consente di utilizzare "*" con lo stesso significato di "#?", cioè per una qualunque stringa di caratteri, come sotto Unix. Inoltre, mette a dispozione le classi di caratteri: "*.[co]" corrisponde a tutti i file che terminano con ".c" o ".o" (non ".co"). E' possibile anche indicare una gamma di caratteri, come in [A-Z1-9] (chi conosce Unix comincerà a intravedere le regular expression di grep). Dulcis in fundo, premettendo "~" al wildcard si otterranno tutti i file che non corrispondono al pattern dato. Queste convenzioni sono valide per tutti i comandi ARP, e non solo per una ristretta gamma come List o Dir. "Type *" listerà tutti i file nella directory corrente, e con un'opzione si può chiedere che ogni file venga intestato col suo nome e separato dal successivo con un avanzamento pagina. Vai a nasconderti, AmigaDOS! Una particolare flessibilità è riservata, inoltre, a Copy e Rename: mediante costrutti del tipo "copy <pref>*<post> <pref2> *<post2>" è possibile eseguire complesse conversioni; ad esempio, "copy T*.c T2*.bak" copierà "TWrite.c" in "T2Write.bak". Il comando aggiuntivo Move permette di usare questo sistema per rinominare file anche attraverso diverse periferiche (nel qual caso esegue un copy e un delete). Un particolare: se durante la copia doveste incontrare un disk error (mai successo? che fortuna!), questi comandi non vi pianteranno completamente in asso, bensì chiederanno gentilmente se è il caso di cancellare dalla destinazione il file incompleto e, poi, se devono comunque continuare la copia. Ogni volta che ha senso, ARP permette un numero indefinito di argomenti: per eseguire quindici assegnazioni logiche, sotto AmigaDOS dovete chiamare quindici volte il comando Assign. Sotto ARP, basterà "Assign <Ass1> <Dir1> <Ass2> <Dir2> ...", e lo stesso si può dire di MakeDir, Delete, Mount e così via. Altro piccolo grande passo in avanti: con il carattere "." si può indicare la directory corrente. E' semplicemente incredibile che questo non si possa fare sotto AmigaDOS. Ogni comando ARP, inoltre, dispone di un help primario, in formato identico ad AmigaDOS, che viene richiamato usando l'argomento "?"; dando però nuovamente "?" dopo l'uscita dell'help si ottiene un'altra descrizione del comando, molto più leggibile. Scendendo nei particolari, credo che alcuni comandi meritino una nota a parte. Prima di tutto Cmp, che permette di confrontare due file binari e di scoprire se sono uguali o diversi: grazie al codice di ritorno, diverso nei due casi, è possibile utilizzarlo nei batch file per eseguire dei controlli. Dir permette di visualizzare solo i nomi dei file o quelli delle directory, eventualmente in reverse. Type, Sort e Search possono accettare l'input da stdin, anziché da un file su disco, aprendo così possibilità pressoché illimitate di piping. Ad esempio, se Nomi.txt contiene una lista di nomi e cognomi in ordine sparso, la linea "Search Nomi.txt Mario NONUM I Sort I Type opt n" visualizzerà in ordine alfabetico le linee del file contenenti il nome Mario, numerandole. Non male, eh?

AShell

La grande novità introdotta con la release 1.3 è però AShell, una potente shell infinitamente più versatile della versione AmigaDOS (che definire shell è in effetti un po' esagerato). Tutti i comandi riguar-

danti i batch file (come If e Skip) sono incorporati in AShell, riducendo così drasticamente i tempi di esecuzioni (nel caso foste preoccupati dalle dimensioni, sappiate che è lunga meno di 8K). AShell permette, inoltre, di gestire comandi residenti in modo molto più sicuro della Shell Commodore, in quanto ogni programma viene sottoposto a checksum prima di essere lanciato: se non era proprio "puro" o se qualche altro programma "bucato" lo ha danneggiato, AShell si rifiuta di eseguirlo. Come se tutto ciò non bastasse, abbiamo la sostituzione dei comandi e delle variabili di sistema: se vengono incorporati in una stringa, preceduti da $, questa viene sostituita dall'output del programma o dal contenuto della variabile, rispettivamente. Quindi, 'Echo "Oggi è il $(date)"' darà come output "Oggi è il 22-Dec-89" (posto che il vostro clock sia settato correttamente e che usiate il comando il 22 Dicembre 89). Queste potenzialità sono accresciute dal fatto che alcuni comandi aggiuntivi permettono di effettuare un input da tastiera scaricandone il contenuto nelle variabili di sistema, ed eseguire su queste, operazioni come la separazione del nome dal path e così via. Credo che sia molto difficile trovare delle ragioni per non servirsi di ARP. Qualcuno dice che i comandi ARP consumano più memoria di quelli AmigaDOS quando girano, ma si tratta di considerazioni superficiali, dal momento che, per esempio, il Sort di AmigaDOS tende a piantarsi se non gli si dà uno stack di grosse dimensioni. Se questo è "consumare poca memoria", allora sono favorevole ad occupare un'espansione da due mega per fare una Dir. Non è, infine, una cosa da poco il fatto che ARexx disponga di un link per alcune routine di ARP, tra cui fondamentale è quella per il file requester. Senza alcuno sforzo, è possibile aumentare enormemente la flessibilità di programmi "fatti in casa". Concludo su una nota dolente, o forse più che dolente irritata: sembra che la Commodore non abbia preso in considerazione l'inclusione di ARP nella release 1.4. Essendomi più volte pronunciato in questa direzione, non posso che rimanerne amareggiato. Rimane la possibilità (un po' fantascientifica, a dire il vero) che la Commodore abbia aggiustato AmigaDOS meglio di ARP. Se ne riparlerà quando sarà pronta la 1.4.▲

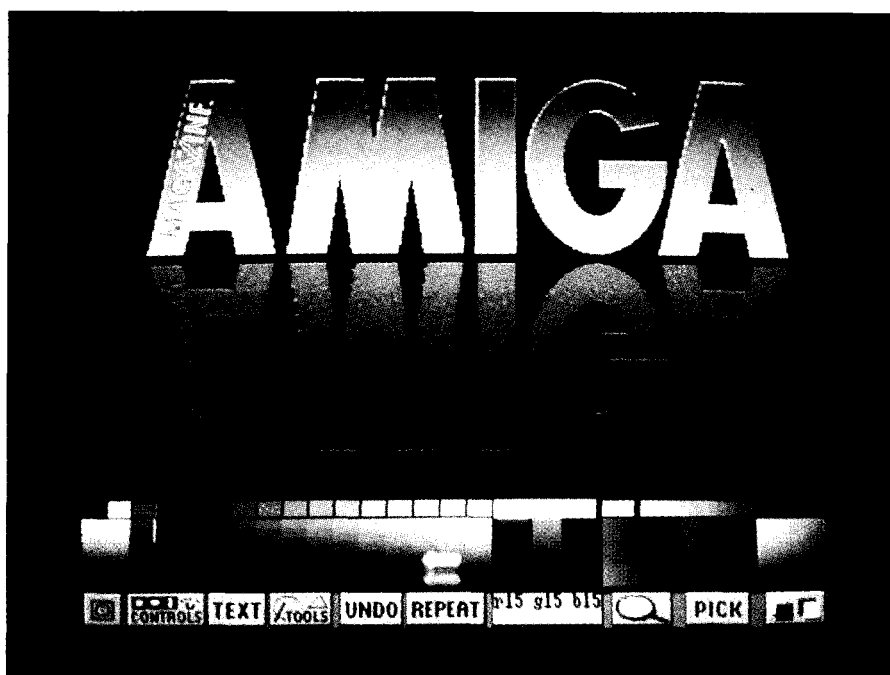
Digi Paint 3

E' l'ultima versione del potente programma della Newtek: Digi Paint 3 in versione PAL

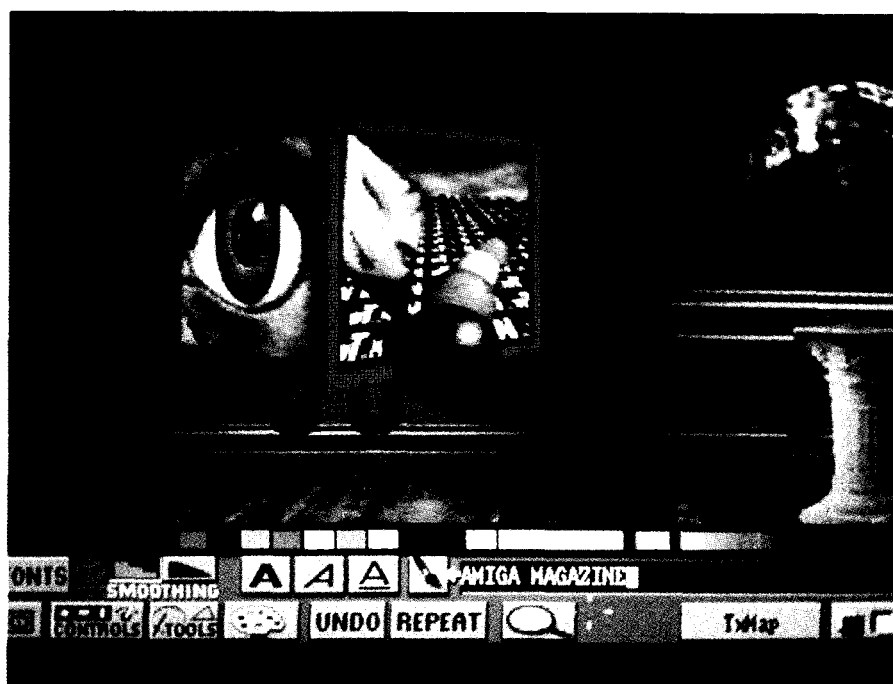
Daniele Cassanelli & Sergio Ruocco

Il mercato del software per sistemi MS-DOS sta ospitando, da qualche anno a questa parte, una lotta senza esclusione di colpi tra la Borland, che sin dalla commercializzazione del Turbo Pascal si è proposta come valida alternativa proponendo ottimi prodotti a prezzi stracciati, e Software House più blasonate (una comincia per M, un'altra per L, indovinate un po' voi...), costringendole a rincorse affannose e controproposte. Ma questo, come avranno avuto modo di chiedersi i nostri simpaticissimi lettori, cosa c'entra con il tema di quest'articolo? Ebbene, anche i produttori di software per Amiga sono passati dalla pubblicità di singoli prodotti al confronto diretto con i concorrenti: basti pensare a Software house specializzate in compilatori, come Lattice e Manx (C 5.04 e Aztec C Express), in word-processor come New Horizons e MSS (Prowrite 2.5 ed Excellence), o in DTP come Gold Disk e Soft-Logik (Professional Page 1.3 e Pagestream 1.7) che sulle pagine delle riviste d'oltreoceano presentano, da un mese all'altro, nuove release dei propri programmi. Anche il settore dei pacchetti di grafica pittorica è agitato da uno scontro all'ultimo pixel tra due giganti dell'HAM: Photon Paint 2.0 della MicroIllusions e Digi Paint 3 della NewTek, ed è di quest'ultimo che ci accingiamo a parlare.

Descrizione

L'elegante confezione comprende il manuale in inglese, due dischetti,

un foglietto di aggiornamento al manuale, un depliant del DigiView Gold (è in commercio ormai da un mese la versione 4.0 del software di gestione che permette di digitalizzare e visualizzare immagini in modo pseudo-HAM in alta risoluzione, cioè 640 x 512 pixel in 4096 colori!) e la cartolina di registrazione clienti che permetterà alla Newtek di sommergervi di depliant informativi di ogni nuova versione dei propri prodotti da qui all'eternità. Confidando nella "amighevolezza" del programma mettiamo da parte il manuale, inseriamo il dischetto programma nel drive e lanciamo il programma. Sul bordo inferiore appare il primo dei quattro menu, denominati ToolBox, attraverso i quali si possono controllare la maggior parte delle numerosissime funzioni del programma, rappresentate da altrettanti gadget grafici realizzati nientemeno che da Jim Sachs, l'autore delle immagini di Defender of the Crown. Le restanti funzioni, come il caricamento e salvataggio delle immagini, dei pennelli, dei font e la selezione dei modi di disegno sono richiamabili dal classico menu a tendina che appare premendo il bottone destro del mouse. Da ognuno dei quattro menu "iconici" (Canvas, Text, Controls e Palette) è possibile impostare il colore corrente della penna, il range di sfumatura per i riempimenti, annullare o ripetere l'ultima operazione eseguita, ingrandire una parte dell'immagine selezionata e passare direttamente ad un altro menu.

Menu Canvas

Esaminiamo ora in dettaglio le funzioni specifiche di ogni menu, cominciando dal Canvas: da qui possiamo impostare una primitiva geometrica (tra retta, spezzata, rettangolo, cerchio, ellisse) o il modo "a mano libera", attivare il riempimento automatico e scegliere la grandezza (ce ne sono sette a disposizione) e il tipo del pennello (tra punto, quattro segmenti variamente orientati, un quadratino e un cerchietto). Spostando il puntatore dall'area occupata dal menu a quella di disegno cominciamo a pasticciare. Selezionando la forbice si può ritagliare una parte del disegno, di forma non prefissata ma corrispondente alla primitiva geometrica selezionata, e trasformarla in un pennello che possiamo sottoporre alle raffinate elaborazioni del menu Controls.

Menu Controls

Quest'ultimo è diviso in tre sezioni: nella prima a partire da sinistra troviamo due gadget, dithering e smoothing, che ci permettono rispettivamente di ottenere tinte intermedie accostando colori simili, simulando così una palette più vasta e di addolcire i profili di aree caratterizzate da colori contrastanti. Il super-gadget Tiles replica in uno spazio di dimensioni e forma a nostra scelta il contenuto del pennello fino a 9 x 9 volte: utilissimo per realizzare superfici composte da elementi regolari (ad esempio, pavimenti, carte da parati, scacchiere ecc.). Due dei tool più potenti e versatili di questo programma chiudono il menu Controls: con il primo (Warp) possiamo arrotolare un pennello lungo un asse orizzontale (o verticale) o attorno ad un punto regolando finemente l'intensità della curvatura mentre dal secondo (Transparency) impostiamo le modalità di trasparenza del pennello e di riempimento delle figure solide con l'opzione Range. Le regolazioni a nostra disposizione sono: grado di trasparenza del punto di luce (Hot Spot) e dell'orlo del pennello, tipo del punto di luce (tra: nessuno, puntiforme, verticale e orizzontale) e suo posizionamento. I tool Tiles, Warp e Transparency sono attivi solo in congiunzione all'opzione TxMap e lavorano su di una copia del pennello corrente; quando TxMap è attivata tutto quello che disegnamo sullo schermo (compresi i caratteri) sarà costituito dal contenuto del pennello che stiamo "smanettando" in quel momento.

Menu Text

Dato sfogo alle nostre capacità artistiche attiviamo il menu Text per firmare il frutto delle nostre fatiche: da qui possiamo scegliere il font con cui vogliamo scrivere, i relativi attributi (tra neretto, corsivo e sottolineato), attivare lo smoothing (se desideriamo attenuare la scalettatura delle diagonali) e, infine, inserire il testo da visualizzare in un area dedicata.

Premendo return il testo viene trasformato in un pennello che può essere comodamente posizionato sullo schermo o ulteriormente elaborato dal menu Controls.

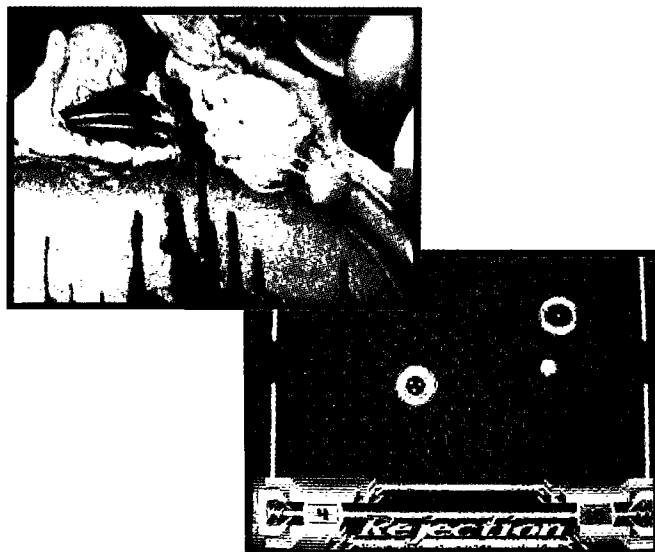
Menu Palette

Un programma HAM degno del formato che supporta deve permettere nel modo più agile il controllo della palette e la scelta della sfumatura appropriata tra le, letteralmente, migliaia di cui questo originale modo grafico è capace; in Digi Paint 3 è stato dedicato un intero menu a questo compito e denominato "Palette". Sostanzialmente ci sono ben cinque modi diversi per navigare con disinvoltura nel mare dei colori e pescare quello giusto: sceglierlo tra i sedici della palette di base, tra una gradazione di questi e il nero o ancora una sfumatura di due colori scelti come estremi (analogo alla funzione RangeTo di Paint), generarlo regolando i classici tre potenziometri a slitta relativi alle componenti rossa, verde e blu (slider), cercarlo in una delle tre aree multicolori dove variano due componenti per volta in accordo ai movimenti del mouse o prenderlo direttamente dal disegno (funzione Pick).

ARexx

Grazie alla compatibilità ARexx, Digi Paint 3 è completamente pilotabile da un programma esterno che gli invii in una sorta di "casella postale" dei comandi; comandi che vengono eseguiti da Digi Paint 3 come se fossero stati impartiti dall'utente per mezzo di mouse e tastiera.
L' "indirizzo" di questa casella è il nome stesso del programma: se rinominiamo Digi Paint 3 in Pippo, ebbene, la casella si chiamerà proprio Pippo! Sul disco del programma è compreso un comando CLI, denominato hey, utile per sperimentare questa interessante tecnica. Ad esempio, per far eseguire a Digi Paint 3 un Undo basta scrivere: >Hey Pippo Undo. E' facile immaginare quali applicazioni possa avere questa caratteristica: complesse e collaudate sequenze di operazioni attivabili con un paio di colpi di mouse risparmierebbero un sacco di tempo altrimenti perso in tentativi infruttuosi.

Transfer 24

Il secondo dischetto contiene Transfer 24, una comoda utility di conversione di formati grafici e di elaborazione immagini (non al livello di Pixmate o Butcher, però). Questo programma può caricare le immagini digitalizzate nel formato privato Digi View o una qualsiasi immagine IFF, modificare risoluzione, luminosità, contrasto, saturazione e nitidezza di un immagine, ridurre il numero dei colori sostituendoli con dei retini e, infine, salvare il tutto in formato standard IFF.

Conclusioni

L'avere a portata di mano, pardon, di mouse, una tale messe di effetti (tra l'altro combinabili liberamente tra di loro aumentando in modo vertiginoso il numero di varianti) da non poter nemmeno essere descritti completamente nel manuale del programma può disorientare almeno in un primo momento anche l'utente più smaliziato ed è necessaria una fase di acclimatamento per prendere progressivamente confidenza con il programma. Bisogna dire, inoltre, che il programma necessita almeno di 1 Mb per girare disinvoltamente, anche se il manuale afferma che bastano 512 Kb; in questo caso, però, risulta impossibile ritagliare un pennello di dimensioni superiori a circa un quarto di schermo e la funzione di undo è disabilitata. Quest'ultimo, però, non è da considerarsi un difetto del programma, che fa già cose egregie in mezzo Mega, ma piuttosto un avvertimento per chi dovrà utilizzarlo intensamente in condizioni non ottimali. A chi possedesse un'eventuale scheda acceleratrice come la A2620, recensita già su queste pagine, potrà far piacere sapere che il programma è stato ottimizzato per il processore 68020, rimanendo però perfettamente compatibile con il buon vecchio 68000 che equipaggia la stragrande maggioranza degli Amiga sul mercato. Digi Paint 3 è ora il dominatore del campo paint in formato HAM, soprattutto se affiancato dal suo naturale complemento, il meraviglioso occhio elettronico di Amiga: Digi View. Siamo curiosi di sapere quale sarà la risposta della concorrenza. ▲

ON DISK è una rubrica mensile di quattro pagine che possono anche essere staccate e conservate, in queste pagine sono descritte tutte le informazioni dei programmi inclusi nel disco, complete di istruzioni, trucchi ecc... In questo spazio troveranno posto giochi, utility e tutto ciò che può fare Amiga.

• Games

Mosaic

Mosaic è un gioco di strategia che permette di confrontarvi con il vostro meraviglioso Amiga.
Lo scopo del gioco è di posizionare, in ordine crescente, delle tessere numerate prima che ci riesca il vostro avversario.

Come si gioca

Le tessere del gioco sono 64, numerate da 1 a 64. Ad ogni giocatore vengono assegnate, all'inizio del gioco, dieci tessere scelte a caso fra le 64 disponibili (le tessere del computer sono quelle poste nella parte alta dello schermo).
All'inizio di ogni turno una tessera si trova al centro dello schermo, il giocatore di turno (la cui scritta è evidenziata) deve scegliere con quale delle sue tessere scambiarla, nel tentativo di porre le proprie tessere in ordine crescente; nel caso la tessera al centro non soddisfi il giocatore, può sostituirla con un'altra tessera cliccandoci sopra.
A questo punto il giocatore può ricliccare sulla nuova tessera (in tal caso significa che vuole passare il turno perché anche la nuova tessera non lo soddisfa) o scambiarla con una sua tessera.
Nello scambiare la tessera centrale con una delle proprie, bisogna fare attenzione a non fornire all'avversario una tessera che gli permetta di vincere o di migliorare notevolmente la propria situazione (perché la tessera scambiata con la centrale va a sostituirla e resta a disposizione dell'avversario che può, durante il suo turno, utilizzarla o sostituirla con una nuova tessera).
Alla fine di ogni match il vincitore riceve 50 punti (5 per ogni tessera), mentre il perdente riceve 5 punti per ogni tessera in ordine crescente.
Il primo dei due giocatori che raggiunge i 500 punti vince la partita
I menu permettono di azzerare i punteggi (New Game), di assegnare ad ogni giocatore nuove tessere, ma senza azzerare i punteggi (New Tiles) o di uscire dal programma (Quit).

Othello

Othello è la versione Amiga del famosissimo gioco di strategia Reversi.
In questo gioco i due giocatori (computer e giocatore) posizionano a turno una propria pedina (contraddistinta da quelle dell'avversario per il diverso colore) nella scacchiera.
La pedina viene posizionata solo se tra di essa e un'altra pedina dello stesso colore ci sono delle pedine dell'avversario; tali pedine vengono sostituite con quelle dello stesso colore di colui che ha mosso.
I due giocatori devono, quindi, tentare di avere il maggior numero possibile di pedine del proprio colore sulla scacchiera e di posizionare le proprie pedine in modo da impedire qualsiasi attacco dannoso da parte dell'avversario.
Quando l'intera scacchiera è piena, vince il giocatore che ha il maggior numero di pedine.
Una buona tattica consiste nel posizionare le proprie pedine ai lati della scacchiera o ancor meglio negli angoli; in tale posizione le pedine sono difficilmente attaccabili e rappresentano quindi una utile roccaforte per poter poi convertire intere file di pedine dell'avversario in pedine del proprio colore.

Rejection

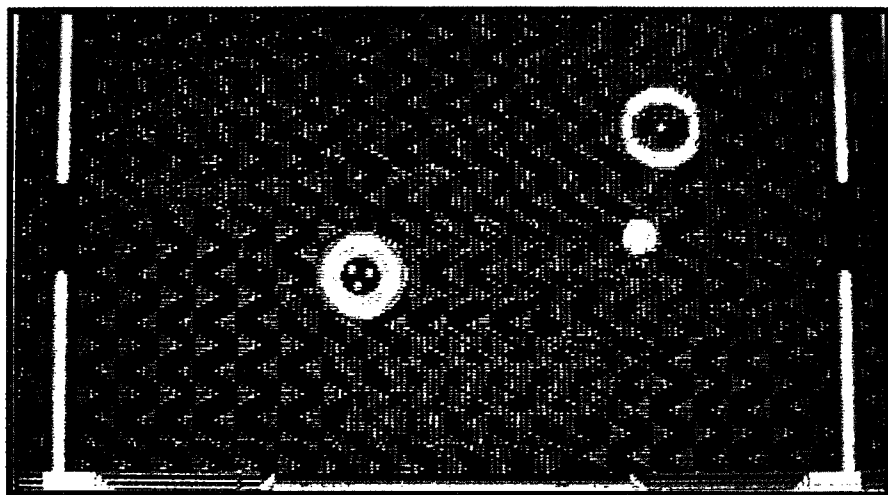
Siete a bordo di una navetta-overcraft, equipaggiata con uno speciale generatore di magnetismo, e dovete cercare di spingere una sfera metallica, che rotola all'interno del campo di gioco, nella porta dell'avversario, sfruttando la forza magnetica emanata dalla propria navetta. La navetta è dotata di otto razzi, ognuno dei quali è attivabile spostando la leva del joystick nella direzione desiderata. L'attrito, grazie al cuscino d'aria, è praticamente nullo e quindi ogni spinta generata dai razzi va ad aumentare o a diminuire la velocità di spostamento della navetta, che, non manovrando alcun razzo, si mantiene costante.
Per fermarsi è necessario utilizzare i razzi opposti alla direzione di movimento. La navetta cambia direzione quando viene a contatto con i bordi del campo o con quella dell'avversario. La palla non può essere colpita perché rotola sul pavimento, mentre le navette galleggiano, grazie al cuscino d'aria a qualche centimetro dal campo di gioco.
L'unico modo per deviare la palla è quello di utilizzare il campo magnetico emanato dalla navetta: normalmente tale campo magnetico respinge la palla, ma tenendo premuto il tasto di fuoco, esso si inverte, attraendola. La forza che agisce sulla palla dipende dalla legge di attrazione tra corpi emananti campi magnetici: più si è vicini alla palla, con più forza la si attrae o respinge. Tenete conto del fatto che non è solo il vostro campo magnetico che agisce sulla palla, ma anche quello del vostro avversario che ne devia la traiettoria.

Come si scelgono i giocatori

Una volta caricato il programma bisogna scegliere il giocatore destro, sinistro e il numero di goal necessari per vincere la partita.
Se volete vedere una partita dimostrativa scegliete per entrambi i giocatori il computer, altrimenti tenete conto del fatto che il giocatore destro deve utilizzare il joystick collegato alla porta 1 (tale joystick viene anche utilizzato per effettuare le scelte dei menu iniziali).Durante la partita la barra spazio blocca il gioco e RETURN permette di continuare. Con ESC è, infine, possibile porre termine alla partita.

• Utility

Macro keys

Macro keys permette di assegnare una parola, una frase, un comando CLI o una serie di comandi a uno dei tasti funzione di Amiga. E' poi sufficiente premere il tasto funzione perché l'intera stringa venga scritta nella finestra attiva.
Macro keys offre, inoltre, il vantaggio di funzionare anche contemporaneamente ad altri programmi, permettendo di utilizzare le proprie macro anche all'interno del proprio Word Processor preferito, a meno che il programma non utilizzi già dei tasti funzione, come capita, ad esempio, con ProWrite che permette di ridefinire solo i tasti funzione da F6 a F10.
E' possibile definire fino a 40 macro (10 macro moltiplicate per le quattro combinazioni possibili: Tasto Funzione oppure Tasto Funzione + CTRL, SHIFT o ALT); ogni macro può essere composta da un massimo di 100 caratteri.
Per ridefinire una macro è sufficiente attivare la finestrella di Macro Keys e premere il tasto o la combinazione di tasti da ridefinire; la macro associata apparirà nella finestra di Macro Keys e sarà possibile ridefinirla utilizzando tutti i tasti di editing di testo, e premendo RETURN per terminare l'editing della macro.
Per inserire il codice carattere corrispondente ad un Return è necessario immettere nella definizione della macro una barra verticale (SHIFT-BackSlash); ciò rende possibile definire anche una serie di comandi in una sola macro: per fare ciò sarà sufficiente inserire alla fine di ogni comando una barra verticale.
Attivando la finestra di Macro Keys appaiono dei nuovi menu, le cui voci sono: on/off, Clear, Load, Save, Help e About.
On/Off serve ad abilitare o disabilitare Macro Keys (il valore attuale appare sulla drag-bar), e quest'opzione si rende molto utile per disabilitare l'utility in caso si stia utilizzando un programma che utilizza i tasti funzione (come ProWrite); l'opzione Clear serve ad azzerare tutte le macro. Per permettere all'utente di disporre di più di un set di macro, l'utility offre la possibilità di caricare o salvare mediante le opzioni Load e Save un set di macro. Il set di macro

Una rettifica per il boot del 68010

Installate un correttore per il 68010 ad ogni reset!

Danny Ross

Installare un 68010 nel vostro Amiga è realmente molto semplice. Si apra il cabinet, si tolga il 68000 (con cura) e si innesti il 68010. E' tutto. La Motorola dichiara un incremento della velocità compresa fra l'8 e il 50% nei confronti di un 68000 e con una piena compatibilità sia per quanto riguarda i pin che il software: con poca spesa (50.000 lire) si ottiene l'equivalente di un 68000 a 10 Mhz.

Naturalmente non è così semplice e, sebbene il 68010 sia effettivamente più veloce di un 68000, i migliori risultati che ho visto (con benchmark standard) si attestano attorno al 5-10%. Il problema sta nel fatto che le funzioni che sono state perfezionate (MULU, per esempio, impiega ben 30 cicli di meno, e DIVU 32) non vengono usate molto spesso (MULU e DIVU quasi mai, relativamente parlando) e così i benefici emergono poco.

Un'eccezione è costituita dal nuovo "Loop Mode", che può velocizzare notevolmente i loop corti, e ciò costituisce un notevole vantaggio per il codice destinato al 68000.

Il problema

Ammettiamo che abbiate deciso di effettuare l'investimento e installato il 68010, che abbiate ormai superato l'irritazione di scoprire che né il "PC Transformer", né l'emulatore del 64 riescono a funzionare, che abbiate incominciato ad usare la vostra macchina, probabilmente vi sarete imbattuti nell'unica crepa del vostro sogno di miglioramento: l'istruzione MOVE SR,<ea>.

Oltre ad aver migliorato il processore per incrementarne la velocità, la Motorola ha aggiunto due nuove prestazioni al 68010. Una è la capacità di gestire la "Virtual Memory" (il 68010 fu il primo microprocessore a incorporarla), l'altra è il supporto per la "Virtual Machine". Ora, mentre il supporto per la memoria virtuale è un'idea piuttosto buona, quello per la "macchina virtuale" ha lasciato perplesse molte persone sin dal momento in cui la Motorola l'ha annunciato.

La questione stava in questi termini: con il 68010 diventava possibile sviluppare nuovi sistemi operativi per macchine non ancora disponibili, facendo girare l'OS in modo utente sotto un sistema operativo per 68010 già esistente ed emulando l'hardware della macchina. In questo modo, ogni tentativo del nuovo OS di accedere all'hardware non ancora disponibile, avrebbe causato una eccezione che il programma emulatore avrebbe potuto trattare.

Siccome un sistema operativo deve, con una certa frequenza, accedere al byte di sistema del registro di stato (se non altro per stabilire il corrente livello di interrupt), i programmi in modo utente non devono essere in grado di leggere tale valore senza che l'OS possa intercettare tale tentativo. Il registro di stato nel nuovo OS sarà quasi certamente differente da quello reale, così la Motorola ha reso l'istruzione MOVE SR,<ea> un'istruzione privilegiata, e ogni tentativo di eseguirla in modo utente genera un'eccezione.

Questo era in gran parte accettabile, i normali programmi utenti non hanno bisogno delle informazioni aggiuntive contenute nel registro di stato supervisore e, in ogni caso, dovrebbero usare l'istruzione MOVE CCR,<ea>. Tuttavia, l'istruzione MOVE CCR,<ea> non era disponibile prima del 68010 e così alcuni programmi per il 68000 di Amiga furono scritti utilizzando l'istruzione privilegiata MOVE SR,<ea>.

Ciò va considerato, adesso, una pratica errata, esistendo una funzione Exec, GetCC( ), prevista proprio al fine di evitare questo problema; ma tutto questo non ci è di grande aiuto con i programmi già scritti. Ciò di cui abbiamo bisogno è un rimedio per il problema MOVE SR,<ea>, e questo è il fine del presente articolo.

La soluzione

Da qualche tempo esiste un programma chiamato DeciGEL che può essere lanciato dal CLI al fine di installare una correzione per il problema del 68010. Questo va già bene, ma non sarebbe più piacevole avere a disposizione la correzione operata da DeciGEL già dalla fase di boot ? La startup-sequence costituisce la risposta più ovvia, ma questo implica editare e copiare file e una notevole quantità di tempo, se avete 200 dischi che desiderate proteggere.

Una soluzione meno onerosa è l'installare la correzione per

il 68010 nel bootblock del disco, ciò di cui abbiamo bisogno è allora un piccolo programma in C simile a quello standard Install, con cui corregere dischi a volontà.

Detto per inciso, la ragione per cui il software per la macchina virtuale non è mai decollato è che è troppo complicato implementarlo: nel momento in cui tutte le routine di emulazione per l'hardware e le istruzioni privilegiate sono state scritte e pienamente testate, la macchina vera e propria sarà probabilmente già pronta; per esempio, si pensi al problema di consentire al nuovo OS di implementare la funzione trace, ricordando che sia il nuovo OS, sia l'applicazione che gira nel suo ambiente dovrebbere essere in modo utente, mentre il sistema operativo emulato "crede" di essere in modo supervisore.

L'installazione di DeciGEL e della startup

Tutti i bootblock Amiga cominciano alla stessa maniera: 4 byte che descrivono il tipo di disco (normalmente "DOS\x00"), 4 byte di checksum, e 4 byte riservati. Questi vanno posti all'inizio del programma:

```
dc.l $444F5300 ; 'D' 'O' 'S' $0
dc.l $F97034C2 ; checksum del bootblock
dc.l $00000370 ; riservati
```

Dopo di ciò ci possono essere fino a 1012 byte di codice che terminino con un normale RTS.

Un programma di bootblock viene chiamato con un puntatore a un Request IO, per un device disco già aperto, in A1 (ciò viene sfruttato molto spesso dai virus) e la base della libreria Exec in A6.
La sua funzione normale è molto scarna e consiste solo nel ricavare un puntatore al codice di inizializzazione della "dos.library" [60-64] (si noti che i valori posti entro le parentesi quadre si riferiscono ai numeri di linea del listato sul disco). Dopo il normale startup siamo liberi di fare ciò che ci piace; così, salviamo tutti i registri [65] e cominciamo con il nostro codice.

Il principio di DeciGEL è molto semplice.
Quando avviene una eccezione causata da una PrivTRAP (violazione di privilegio) controlla l'istruzione che l'ha generata per verificare se si tratti di una MOVE SR,<ea>, e, se lo era, la si tramuta in una MOVE CCR,<ea>, tentando di eseguirla nuovamente.

Organizzazione dello stack delle eccezioni nel 68010 (corto)

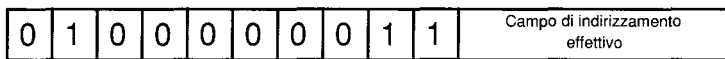
Al momento dell'eccezione

$06 (SP)	Registro di stato	
$04 (SP)	Program Counter (word bassa)	
$02 (SP)	Program Counter (word alta)	
(SP)	Formato	Offset del vettore

Dopo il caricamento dei registri

$0E (SP)	Registro di stato	
$0C (SP)	Program Counter (word bassa)	
$0A (SP)	Program Counter (word alta)	
$08 (SP)	Formato	Offset del vettore
$06 (SP)	D0.L (word bassa)	
$04 (SP)	D0.L (word alta)	
$02 (SP)	A0.L (word bassa)	
(SP)	A0.L (word alta)	

Fig. 1

L'installazione di DeciGEL avviene richiedendo ad Exec un po' di memoria per l'handler dell'eccezione (in quanto questo codice deve rimanere in memoria dopo che il bootblock ha concluso il suo compito).
Poi copia il codice dell'handler dell'eccezione [92-103] in questa memoria e, infine, lo aggancia al vettore delle eccezioni nella posizione corrispondente alla violazione di privilegio.

Quando avviene una PrivTRAP sul 68010, vengono salvati sullo stack vari registri (si veda la figura 1) e viene chiamato

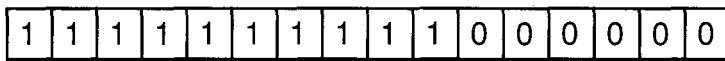



Fig. 3

MOVE SR, <ea> | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | Campo di indirizzamento effettivo

Maschera per l'esclusione del campo <ea> | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0

= $40C0

Fig. 2

il codice indirizzato da PrivTRAP. Il nuovo handler salva i due registri che intende utilizzare (si veda ancora la figura 1) e poi ricava l'istruzione dal valore che aveva il PC (Program Counter) al momemto dell'eccezione.

Siccome il codice dell'istruzione MOVE SR,<ea> non ha un valore fisso, a causa del campo <ea>, DeciGEL ne maschera il contenuto (si veda la figura 2) per ricavarne il puro codice dell'operazione, che risulterà indipendente dal registro utilizzato ($40C0 nel caso di MOVE SR,<ea>). Se il risultato non è $40C0, DeciGEL presume che l'eccezione PrivTRAP sia stata causata da qualche altra istruzione e, dopo aver restaurato tutti i registri, chiama il vecchio handler. Viceversa, se trova il valore $40C0, l'istruzione viene convertita in MOVE CCR,<ea>, impostando il nono bit nella parola d'istruzione (si veda la figura 3), e l'istruzione viene rieseguita.

I veterani del 68000 possono aver notato qualcosa di bizzarro nel diagramma di figura 1.
Dopo una normale eccezione, il 68000 è solito salvare solo il registro di stato e il PC corrente, ma il 68010 richiede qualche volta maggiori informazioni e ha un'organizzazione differente dello stack.

Per distinguere tra i due tipi di stack (in modo che l'istruzione RTE possa ricavare le informazioni giuste prima di ritornare) il 68010 carica una word ulteriore sullo stack all'inizio del processo di eccezione. Parte di questa word viene detta "campo formato".

Un valore di %0000 (binario) indica uno stack corto (4 word, come in figura 1) e %1000 indica uno stack lungo (29 word, usate per lo più per il supporto della memoria virtuale).

Ciò può sembrare abbastanza irrilevante, ma significa che se il programma Boot68010 viene fatto girare su una macchina dotata di 68000 e avviene una PrivTRAP, si scatenerà il caos perché il PC verrà ricavato da una posizione errata dello stack.

Fortunatamente, è altamente improbabile che il dato a questo PC risulti proprio $40C0 e DeciGEL lo riconosca come un PrivTRAP da elaborare. Sfortunamente, è piuttosto probabile che il presunto PC sia dispari, e che l'istruzione MOVE.W alla linea [94] causi un errore di indirizzo. Sembra ci sia qui dello spazio per migliorare il programma.

Visualizzazione del messaggio e conclusione

Se Boot68010 si concludesse a questo punto, certamente realizzebbe il suo scopo compiutamente. Ma, come potremmo sapere che la correzione per il 68010 è stata installata, e che cosa ci dimostrebbe che questo è un bootblock utile e non un virus ?

Per rendere il programma più interessante, viene visualizzata una piccola immagine (256 x 16 pixel) al centro dello schermo, mentre si resta in attesa che l'utente prema il tasto del mouse.
Il colore dell'immagine varia nella gamma dei blu per renderla più interessante.

Per fare questo, il programma prima apre la "graphics.library" e localizza il puntatore alla Copperlist iniziale [113-119].
Poi, imposta DDFSTRT e DDFSTOP.
Questi valori vengono usati per indicare all'hardware che gestisce il video quando (in termini di posizione orizzontale del pennello elettronico) deve cominciare a caricare i dati dei bitplane e quando deve terminare. Potete mutare il valore e vedere il messaggio muoversi verso sinistra e verso destra, ma assicuratevi di cambiare entrambi i valori della medesima quantità.

Se la differenza tra i due diventa troppo grande (o troppo piccola), l'hardware video caricherà troppi (o troppo pochi) dati per ciascuna riga e l'immagine andrà fuori squadra.
Inizializzati i registri video, il programma chiede ad Exec della memoria CHIP in cui copiare l'immagine e la Copperlist che permetterà di generale il display.
Dopo aver controllato se AllocMem è riuscita ad allocare memoria sufficiente per la Copperlist e la bitmap [133], il programma procede generando la Copperlist.

La prima parte della Copperlist appare così:

`WAIT (0,128)`
- il pennello elettronico è nel punto più alto dello schermo

`MOVE ($0000,BPLCON0)`
- disabilita ogni attività relativa ai bitplane

`WAIT (135,0)`
- posizione iniziale dello schermo

`MOVE ($1000,BPLCON0)`
- imposta bassa risoluzione e un bitplane

`MOVE ($????,BPL1PTH)`
- word alta dell'indirizzo del bitplane

`MOVE ($????,BPL1PTL)`
- word bassa dell'indirizzo del bitplane

L'indirizzo del bitplane viene calcolato a partire dalla base della memoria CHIP appena allocata e inserito nella Copperlist [164-167].

Poi, un breve loop genera 16 linee di Copperlist. Queste sono le istruzioni per il copper che modificano i colori dell'immagine; così, in effetti, un'immagine con un solo bitplane appare con 16 colori diversi.

Le istruzioni per il copper sono di questo tipo:

`WAIT (135+n,0)`
- inizio di ogni linea del display dell'immagine

`MOVE (colore,COLOR01)`
- imposta il colore per questa linea

Le ultime due istruzioni per il copper [157-158] chiudono tutta l'attività dei bitplane e terminano la lista con una WAIT impossibile. Il pennello elettronico raggiungerà il vertical blanking prima che la WAIT possa essere soddisfatta e il copper sarà costretto a far ripartire la lista di istruzioni dall'inizio. Ora tutto quello che resta da fare è copiare l'immagine nella memoria CHIP [160-162] e costringere il copper a iniziare l'esecuzione della nuova lista. Ciò si realizza caricando in COP1LCH l'indirizzo della nuova lista e facendo un qualsiasi accesso a COP1JMP.

Il resto del bootblock altera ripetutamente la Copperlist in modo che i colori usati nella istruzione MOVE vengano gradualmente incrementati.
Compare un'istruzione AND al fine di limitare la quantità del cambiamento e di far ricominciare da capo il cilco dei colori; così l'effetto ottenuto è quello di una gamma di blu che fa pulsare ritmicamente l'immagine, fino a quando non venga premuto il pulsante sinistro del mouse [193].

L'immagine (dovrete far girare il programma per sapere che cosa vi appare... ma non aspettatevi granché) è stata creata con DPaint II, salvata come una semplice bitmap mediante un programma di traduzione dal formato IFF a quello delle bitmap e poi è stato fatto un dump esadecimale per assembler del file della bitmap. Non dovreste incontrare grosse difficoltà nel crearvi un'altra immagine, ma ricordate che, con il codice così com'è, deve essere esattamente di 256 x 16 pixel.

Snellire il codice

Nel programma ho usato ogni trucco da me conosciuto per mantenere il codice il più breve possibile. Questo può sembrare senza motivo visto che la lunghezza finale del programma è di 910 byte, ma un programma compatto è un programma più veloce (di solito) e, in ogni modo, 114 byte in più costituiscono uno spazio adatto per eventuali migliorie (si veda il paragrafo finale).

Questa è una breve guida per risparmiare memoria, destinata ai novizi del 68000:

- si usi sempre MOVEQ quando è possibile
- si usi BSR invece di JSR
- si usi sempre BRA.S (diramazione corta), se possibile
- si evitino le costanti lunghe, in quanto occupano un'enorme quantità di memoria
- si usino i registri quanto più è possibile, sono molto veloci, (la memory è molto lenta) e il codice risulta più compatto
- il codice con indirizzamento relativo al PC (del tipo LEA nome(PC), A1 invece di LEA nome,A1) è sia indipendente dalla posizione (una necessità per il codice di un bootblock), sia 2 byte più corto del codice con indirizzamento assoluto (di solito)
- si cerchino le ripetizioni di indirizzamenti indicizzati o con spiazzamento. Per esempio:

```
MOVE.W $46(A0),D0
```

può spesso essere rimpiazzato da:

```
LEA $46(A0),A4      ;usato una sola volta

MOVE.W (A4),D0      ; poi si usi questo ogni
                    volta,

...                 ;invece dell'istruzione

MOVE.W (A4),D0      ; indicizzata o con
                    spiazzamento

...
```

I modi con indirizzamento complesso impiegano una grandissima quantità di tempo per essere risolti e incidono piuttosto pesantemente sulla memoria. Si veda [87-89] per un esempio estremo di quanto abbiamo detto.

Il programma Install

Scritto in C, il programma Install è un tipico programma di utilità, funziona, ed è esattamente quello per cui è stato progettato.
Avrei potuto scriverne uno migliore, ma non ce n'era motivo, visto che questo articolo è dedicato alla programmazione del bootblock e non al C. Tuttavia, conclusa l'abiura, vediamo ciò che fa.

Install68010 prende un dump esadecimale del programma Boot68010 e lo scrive, ripetutamente, nei settori di boot del drive specificato. Non sono richiesti parametri (né accettati), e l'aggiunta di una stringa dopo il nome del programma causerà la visualizzazione di un piccolo messaggio sul programma stesso: che cosa fa e perché.

Nella prima fase della sua esecuzione, Install68010 alloca memoria per il buffer del disco (nella memoria CHIP), copia il codice di Boot68010 nel buffer del disco e crea un blocco di richiesta IO al fine di usare il "trackdisk.device" per scrivere direttamente sul disco. L'utente è poi interrogato per sapere quale drive va utilizzato, e quando viene ricevuta una risposta accettabile, viene aperta la corrispondente unità del "trackdisk.device" (l'unità 0 corrisponde a "df0:", l'unità 1 a "df1:" e così via).

Nella seconda fase attende che l'utente inserisca il disco da installare, controlla se quel disco è protetto dalla scrittura e, se non lo è, scrive il codice di boot nei settori 0 e 1 (i settori di boot). Il controllo degli errori è piuttosto limitato e sembra che a volte qualcosa vada storto senza che il programma riesca ad accorgersene.
Il risultato è che, sebbene Install68010 faccia ritorno con un messaggio di OK, il disco non viene installato. Non ho alcuna idea sul perché questo avvenga, ma invariabilmente l'errore scompare se si estrae il disco e lo si reinserisce. E' probabilmente dovuto a qualcosa che ha a che fare con il contatore del cambiamento di disco, qualcuno può fare un po' di luce ?

Di fatto, per usare Boot68010 basta compilare Install68010 e lanciarlo da CLI, ma, ovviamente, in questo modo non potrete personalizzare il codice. Per ottenere i migliori risultati, occorrerebbe possedere un programma capace di produrre dump esadecimali adatti al C ("0x" invece di "$" per i valori esadecimali) e uno per calcolare il checksum del bootblock.

Migliorie e conclusioni

Boot68010 potrebbe essere perfezionato in molti modi. In primo luogo, si dovrebbe controllare se la macchina è un 68000 e non un 68010 per evitare errati riferimenti allo stack, come descritto in precedenza. Poi, siccome esiste molto spazio libero (114 byte), si protrebbero aggiungere altre funzioni.
Per esempio, un allocatore di memoria FAST (per coloro che possiedono una macchina con un Megabyte o più di memoria e hanno problemi con il particolare programma presente sul disco) o un po' di musica o un menu di boot (forse è un po' difficile in questo spazio).

Anche Install68010 potrebbe essere migliorato. Dovrebbe essere in grado di effettuare checksum del bootblock in maniera automatica (come "additive carry wrap around sum of 0xFFFFFFFF" stando a quanto dichiara la Commodore), risparmiandoci, così, un mucchio di seccature, e non dovrebbe confondersi, come qualche volta gli capita.
Un'altra miglioria potrebbe essere quella di permettergli di leggere il codice di boot dal disco, senza dover fare un dump esadecimale.

Tutte le migliorie citate sono basate sui due programmi di questo articolo.

Vi sono naturalmente molte più idee e usi per i programmi di bootblock: decompattatori per boot e correttori di disco per bootblock molto lunghi, programmi di installazione che prendono il codice eseguibile e creano un disco capace di lanciarlo dal bootblock, menu di boot, e boot dimostrativi, tutto un insieme di cose che può essere (e certamente è stato) tentato.

Fornendo un intero Kbyte da programmare, la Commodore ha aperto il campo ad un vasto frutteto di miniprogetti e programmi di utilità.

I moduli residenti di Exec

Un insieme di funzioni in C per penetrare nel cuore di Exec

Romano Tenca

Uno degli aspetti meno conosciuti della libreria Exec è costituito dai moduli residenti. Essi offrono però grossi servigi a chi sappia utilizzarli, perché permettono di inserirsi in profondità nel sistema, sfruttando a fondo le potenzialità di Exec. Il potere messo a disposizione dai moduli residenti finora è stato usato, in tutta la sua estensione, da pochi programmi, per quanto mi è dato di sapere: in primo luogo si può ricordare che una intera generazione di virus (ancora loro), che fa capo a Byte Bandit, ne ha fatto largo, quanto improprio, uso; in secondo luogo, si possono segnalare il ben più utile "ramdrive.device" del WorkBench 1.3 e uno dei migliori programmi del genere "Viruskiller" e "Guardian" dell'italiano Leonardo Fei.

Come si è, forse, già notato, si tratta, in ogni caso, di software dotato delle medesime caratteristiche: la capacità di sopravvivere a un reset della macchina, quando questo non comporti il distacco della corrente elettrica, e la capacità di farsi lanciare (o perlomeno inizializzare) automaticamente dal sistema operativo ad ogni nuovo reset.

Sono queste, fondamentalmente, le due maggiori potenzialità dei moduli residenti: esse permettono, di fatto, di aumentare virtualmente la memoria ROM a disposizione del sistema ad ogni nuovo reset, utilizzando al suo posto una parte della memoria RAM e il codice o i dati ivi contenuti. Vediamo dunque di capire come è stato programmato il sistema operativo per svolgere queste due distinte operazioni. Precisiamo subito che le osservazioni che faremo sono state verificate con la versione 1.2 del sistema operativo che corrisponde alla versione 33.192 della libreria Exec. Tali osservazioni dovrebbero valere anche per le versioni 34 di Exec corrispondenti alle versioni 1.2.1 e 1.3 del Kickstart.

Esse non valgono per le versioni anteriori alla 1.2 e possono anche generarsi dei problemi con gli ultimi modelli di Amiga che montano 1 Mega di CHIP RAM, come chiariremo meglio in seguito.

Come far sopravvivere la memoria

Per capire meglio il modo in cui si può proteggere la memoria dal reset, vale la pena di esaminare, in maniera sintentica, alcune fasi della procedura di reset.

Ogni volta che avviene un reset della macchina, sia che questo sia stato causato dalla pressione dei tasti Control+AmigaSinistro+AmigaDestro, sia che sia stato provocato da una "Guru meditation", il sistema operativo controlla che l'elenco della memoria libera non sia corrotto. Tale elenco è costituito da una serie di strutture MemHeader collegata alla lista contenuta in ExecBase, sotto il nome MemList.

Se la lista viene ritenuta corretta allora si passa ad esaminare il valore KickMemPtr, sempre di ExecBase. E' di un puntatore a una lista di strutture MemList collegate in maniera monodirezionale. Questo aggregato è così definito in "exec/memory.h":

```
struct MemList {
   struct Node ml_Node;
   UWORD ml_NumEntries;
   struct MemEntry ml_ME[1];
};
```

Si tratta dunque di una struttura Node Exec, seguita da un numero variabile di strutture MemEntry, la cui quantità viene espressa dal valore ml_NumEntries. Quando la si intenda usare con un numero di MemEntry superiore a 1, converrà definire una nuova struttura che contenga esattamente il numero di MemEntry che servono.
La struttura MemEntry viene definita, sempre in "exec/memory.h", a questo modo:

```
struct MemEntry {
   union {
      ULONG meu_Reqs;
      APTR  meu_Addr;
   } me_Un;
   ULONG me_Length;
};
```

E' un aggregato composto da due longword in cui la prima corrisponde alla union me_Un e la seconda al campo me_Length. Della union a noi interessa il secondo valore: il campo me_Un.meu_Addr, che esprime un indirizzo di memoria. Esso può essere anche espresso dall'abbreviazione me_Addr. La seconda longword (me_Length) rappresenta, invece, la lunghezza di tale area di memoria.

In fase di reset, tutta la memoria indirizzata dalle MemList collegate a KickMemPtr non verrà ritenuta libera e non sarà messa a disposizione del sistema. Ciò che è contenuto in esse, sopravviverà, dunque, al reset.

Ovviamente, se, in caso di "Guru", un programma avesse per caso danneggiato queste locazioni di memoria, esse non saranno certamente riparate, né saremo avvertiti in alcun modo di tale evento. Exec esercita un controllo solo sulla memoria libera e sulle proprie variabili, non sul contenuto della memoria protetta. Converrebbe, quindi, effettuare un checksum per controllare che la memoria protetta non sia stata corrotta.
In pratica, per preservare delle regioni di memoria dal reset, basta costruire una MemList, le cui MemEntry localizzino le diverse aree di memoria, e agganciarla a KickMemPtr, ponendo nella variabile della struttura Execbase un puntatore alla nostra MemList. Sia detto una sola volta: quando si modificano valori della struttura ExecBase è necessario delimitare il codice con le funzioni Exec Disable( ) (offset $ff88) ed Enable( ) (offset $ff82): con ciò vengono inibiti gli interrupt e il multitasking, pertanto il codice eseguito fra le due funzioni deve essere molto breve e veloce.

Bisogna anche preoccuparsi di salvare il valore già presente in KickMemPtr, ponendolo nel campo MemList.ln_Node.ln_Succ, evitando così di annullare altre richieste come quella eventualmente effettuata dal device RAD. Bisogna ricordarsi, inoltre, che la nostra MemList deve, a sua volta, risiedere in un'area di memoria protetta e quindi situarsi fisicamente in un'area individuata dai parametri di una delle sue MemEntry.

Infine, occorre ricalcolare il valore della variabile KickCheckSum presente in ExecBase. Questo è indispensabile, altrimenti Exec riterrà il sistema corrotto e non prenderà in considerazione il valore contenuto in KickMemPtr. Ciò si realizza comunque con estrema semplicità, attraverso la funzione Exec SumKickData( ) (offset $fd9c) e un'istruzione C di questo tipo:

```
SysBase->KickCheckSum=SumKickData();
```

Un'ultima considerazione va fatta a proposito del tipo di memoria da utilizzare: non tutta la memoria può essere protetta, specie quella delle espansioni di memoria. Ciò avviene perché la memoria aggiunta mediante espansioni, viene collegata al sistema attraverso le funzioni della libreria "expansion.library", la quale entra in funzione solo in un successivo momento delle operazioni di reset. Durante la fase in cui Exec si accinge a proteggere la memoria, le espansioni non risultano ancora disponibili e, quindi, il tentativo di protezione non può avere successo.

Si può proteggere la CHIP RAM e la memoria compresa tra i valori $C00000 e $D80000; come si può notare, vi è compresa l'espansione di memoria interna di Amiga 500 che è situata tra $C00000 e $C80000 e, quindi, in un modello con 512K di CHIP RAM e 512K di espansione interna, ogni locazione di memoria può essere protetta. Un problema può forse insorgere con i nuovi modelli di Amiga dotati di un Mega di CHIP RAM (si tratta di quelli già predisposti per i nuovi chip custom): anche "RAD:" su questi modelli richiede che si lanci, durante la startup-sequence, il comando SetPatch con l'opzione -r, la quale modifica il valore ColdCapture di ExecBase correggendo la routine di reset. Pare, inoltre, che il problema verrà risolto con la versione 35 del kickstart. Comunque, non possedendo tale tipo di macchina, non ho potuto verificare queste affermazioni.

Per eliminare la protezione dalla memoria, basta rimuovere la nostra MemList dalla lista di KickMemPtr e ricalcolare KickCheckSum: al prossimo reset la memoria verrà considerata del tutto libera. Si noti, però, che fino a quel momento essa resta a tutti gli effetti memoria "occcupata": se si vuole restituirla al sistema occorre chiamare FreeMem( ) o FreeEntry( ).

L'inizializzazione automatica e la struttura Resident

Oltre ad offrire la possibilità di proteggere la memoria, Exec permette anche di lanciare automaticamente, al momento del reset, il codice che vi abbiamo salvato. A tal fine occorre indicare al sistema la locazione di memoria a cui cedere il controllo, cioè la funzione utente da eseguire.

Per fare questo si deve usare la struttura Resident, che viene definita in "exec/resident.h". Questo aggregato, chiamato anche romtag, serve, in generale, a fornire al sistema tutte le informazioni necessarie per effettuare le diverse inizializzazioni automatiche che avvengono non solo durante le fasi di reset, ma anche in altri momenti dell'attività del sistema operativo: per esempio, hanno un modulo di questo tipo tutte le librerie shared (condivise) o i device exec.
Per un elenco dei moduli, presenti di default nella versione 1.2 (33) del Kickstart, si consulti la tabella 1.
La struttura è così definita in C:

```
struct Resident {
   UWORD rt_MatchWord;
   struct Resident *rt_MatchTag;
   APTR  rt_EndSkip;
   UBYTE rt_Flags;
   UBYTE rt_Version;
   UBYTE rt_Type;
   BYTE  rt_Pri;
   char  *rt_Name;
   char  *rt_IdString;
   APTR  rt_Init;
   };
```

Il primo campo (rt_MatchWord) deve sempre contenere il valore RTC_MATCHWORD ($4afc), che corrisponde a una istruzione illegale nel set del 68000: serve probabilmente come valore di controllo per stabilire la validità del romtag.

Il secondo campo (rt_MatchTag) ha presumibilmente la stessa funzione e deve sempre contenere l'indirizzo della struttura Resident.

Il terzo campo (rt_EndSkip) appare piuttosto misterioso: deve avere anch'esso funzioni di controllo (forse per un eventuale checksum operata dal sistema) e puntare teoricamente al termine del nostro codice. Bisogna però evitare che esso indirizzi una locazione inferiore a quella della struttura Resident o che cada nel mezzo del nostro programma. In generale, si ci può porre al riparo da cattive sorprese attribuendogli un valore corrispondente a un indirizzo di memoria posto subito dopo la struttura Resident.

Il quarto campo (rt_Flags) deve contenere uno o più dei seguenti flag, combinati attraverso un OR logico:
RTF_COLDSTART ($80): indica ad Exec che il modulo va inizializzato durante le fasi di reset e deve essere quindi presente se si vuole che la nostra funzione venga attivata in tale fase.
RTF_AUTOINIT ($01): indica ad Exec che si richiede una inizializzazione di particolari aree di memoria, che verranno automaticamente allocate da Exec. Questo flag viene utilizzato, soprattutto, nelle librerie e nei device per sfruttare le routine Exec di avvio automatico (a tal proposito si possono consultare gli articoli di Steve Simpson apparsi su Transactor per Amiga n.3, 1989, p.68 e n.4, 1989, p.63).

Il quinto campo (rt_Version) contiene il numero di versione del romtag; scorrendo la lista dei romtag presenti nel Kickstart 1.2 si può scoprire che tutti i moduli contengono il valore 33.

Il sesto campo (rt_Type) deve contenere un flag del tipo usato per le strutture Node di Exec, la cui definizione si trova nel file "exec/nodes.h". Se il romtag non ha un particolare valore per Exec si può usare il simbolo NT_UNKNOWN (0); nel nostro caso gli si può preferire il valore NT_BOOTNODE (16), trattandosi di codice da far eseguire durante la fase di reset.

Il settimo campo (rt_Pri) è molto importante: grazie al valore ivi contenuto Exec può stabilire l'ordine con il quale i vari romtag verrano inizializzati dal sistema.
Di default, la priorità più alta (120) è quella della libreria Exec e la più bassa è quella del modulo "strap" (-60) che si incarica di leggere il bootblock del disco per lanciare l'AmigaDos. Nel Kickstart 1.3, subito prima di "strap", compare il modulo "romboot" con priorità -40: esso non esisteva nella versione 1.2 ed è incaricato della riconfigurazione del sistema per gestire le operazioni di boot da device diversi dal disk drive interno.

Perché il nostro codice venga eseguito prima del lancio del Dos occorre prevedere una priorità superiore a -60 (o meglio ancora -40). Si tenga, inoltre, presente che tutti gli altri moduli contenuti in ROM e presenti di default nel sistema hanno una priorità uguale o superiore a 0; pertanto perché il nostro codice possa trovare tutte le librerie e i dispositivi di Exec già inizializzati e pronti all'uso dovrà avere una priorità inferiore a 0. In conclusione tutti i valori compresi tra -1 e -39 sono utilizzabili con la sicurezza di trovarsi in un ambiente in cui tutto il sistema (tranne il DOS) è già stato inizializzato. Nulla vieta, comunque, di stabilire una priorità più alta, specie se si vuole scendere ancora più in profondità all'interno di Exec.

L'ottavo campo (rt_Name) deve contenere un puntatore a una stringa, il cui ultimo carattere sia uno zero, che indichi il nome del modulo. Questo nome è il principale fattore di identificazione del modulo.

Il nono campo (rt_IdString) deve contenere un puntatore a una stringa, sempre terminata da uno zero, che contenga altre informazioni utili all'identificazione del modulo. I programmatori della Commodore usano per i loro moduli lo schema seguente:"nome versione.revisione (giorno mese anno)". Scorrendo, ad esempio, la versione 1.2 (33) del Kickstart, si può scoprire che la libreria Exec contiene questa IdString:"exec 33.192 (8 Oct 1986)". Converrà, per uniformità con i moduli di default, osservare questo schema, anche se ciò non risulta strettamente necessario.

L'ultimo campo (rt_Init) deve contenere un puntatore alla nostra funzione, cioè al codice che verrà eseguito in fase di reset. Se però nel campo rt_Flags è stato previsto il valore RTF_AUTOINIT, il puntatore assume un diverso significato che serve al sistema per individuare tutti i dati relativi al lancio di una libreria o di un device: si rimanda ancora una volta ai due articoli di Transactor per Amiga, citati in precedenza, per maggiori ragguagli ed esempi di codice. Nel nostro caso, basterà porre in questo campo l'indirizzo della funzione che intendiamo sia eseguita durante il reset.

ResModules e KickTagPtr

Exec nel corso del reset si incarica di costruire un elenco di tutti i moduli presenti nel sistema; per far questo costruisce un array di puntatori alle diverse strutture Resident (ordinato secondo la priorità), terminato da una longword posta a 0, e poi colloca l'indirizzo di questo array nel campo ResModules di ExecBase.

Per trovare i diversi moduli riconosciuti e accettati dal sistema basterà, dunque, scorrere questo array. E' anche possibile trovare l'indirizzo di un modulo mediante la funzione Exec FindResident(nome) (offset $ffa0), che ritorna un puntatore alla struttura Resident collegata a ResModules con il nome richiesto, oppure 0, se non viene trovata.

Per aggiungere un nuovo modulo al sistema non si può intervenire direttamente su ResModules, ma occorre passare attraverso un altro campo di ExecBase, precisamente KickTagPtr.

Se questo campo è uguale a 0, significa che non è stato

aggiunto alcun modulo a quelli di default; in caso contrario contiene un puntatore a un array di puntatori a strutture Resident. Tale array può terminare con uno 0 che ne indica la fine, oppure con una longword con il bit 31 impostato. Quest'ultimo valore, una volta che si sia azzerato il bit 31, va interpretato come un puntatore a un altro array di strutture Resident. Anche per questo vale lo stesso discorso, per cui potremmo dire che KickTagPtr è un puntatore a una lista di array collegati attraverso il loro ultimo campo.

Le cose appaiono piuttosto complicate, ma non del tutto a caso. Il problema che questa particolare implementazione risolve è il seguente: per aggiungere un nuovo modulo dobbiamo fornire ad Exec un puntatore alla struttura Resident; sia il nostro codice, sia la struttura Resident, sia il puntatore devono risiedere in un'area di memoria protetta, se vogliamo che sopravvivino al reset: tutti dunque dovranno essere compresi nella memoria indirizzata dalla MemList che andremo ad aggiungere a KickMemPtr (come abbiamo in precedenza spiegato).
La lista di array permette di porre l'array che ci riguarda all'interno della nostra area di memoria protetta e di collegarci da una parte a ExecBase e dall'altra agli altri array di puntatori a romtag, aggiunti in precedenza al sistema.E' così possibile costruire una lista virtualmente infinita di puntatori, che risiedono in aree distinte di memoria protetta.

Praticamente, nel nostro caso, occorrerà in primo luogo definire un array di 2 longword, poste nella memoria protetta; porre nel secondo elemento il valore presente in KickTagPtr con il bit 31 impostato, se diverso da 0, e nel primo elemento l'indirizzo della nostra struttura Resident. Ora è possibile inserire in KickTagPtr l'indirizzo del nostro array così com'è (cioè senza impostarne il bit 31). Per finire occorre ricalcolare KickCheckSum nel modo che abbiamo in precedenza chiarito.

Con ciò il nostro codice è pronto per essere incluso fra i moduli di sistema, ma l'operazione non verrà eseguita immediatamente da Exec, bensì solo al prossimo reset: quando si tratterà di ricostruire l'elenco di ResModules, Exec controllerà KickTagPtr; se questo non è uguale a 0, porrà le strutture Resident aggiuntive, ivi indirizzate, nella lista ResModules e le inizializzerà, secondo le istruzioni che ha ricevuto e nel rispetto della priorità stabilita, lanciando la funzione indirizzata da rt_Init.
Se volessimo eliminare il nostro modulo dal sistema occorrerà distaccare il nostro array da KickTagPtr e ricalcolare KickCheckSum: al prossimo reset il nostro modulo non verrà più compreso nell'array di ResModules.

Il programma d'esempio e le sue funzioni

Spesso i programmi che agiscono nel modo che abbiamo appena descritto vengono scritti in linguaggio macchina, per risolvere in maniera agile il problema di evitare che il Dos, dopo che ci ha lanciati, ci rimuova anche dalla memoria. Questo ostacolo può essere aggirato in molti modi, per esempio allocando della memoria e poi copiando in essa tutti i dati e il codice che deve rimanere residente. Il nostro programma esempio affronta e risolve il problema utilizzando solo il linguaggio C (nell'implementazione Lattice).

Ogni programma eseguibile è composto da uno o più hunk che vemgono caricati dal loader del Dos in aree distinte di memoria. Il primo hunk corrisponde, nel Lattice C, al segmento di codice con cui inizia il programma; esso viene determinato, di fatto, dal primo file indicato sulla linea di comando di blink. Normalmente si tratta del modulo di startup del C, come "c.o"; nel nostro caso, invece, non viene usato alcun modulo di startup e il programma inizia direttamente con la prima linea di codice della prima funzione del nostro sorgente.

A questo modo si perde la possibilità di usare tutta una serie di funzioni standard del C, specie quelle di input e output (non si possono usare cioè funzioni come printf( ), fprintf() e così via), e funzioni, come exit( ), che fanno riferimento a indirizzi esterni contenuti in "c.o"; inoltre non avviene alcuna inizializzazione, nemmeno quella di SysBase (che deve puntare alla base della libreria Exec, se si fa un link con "amiga.lib"). Dunque occorrerà effettuare all'inizio del programma le inizializzazioni che servono (SysBase, DOSBase ...) ed aprire eventuali FileHandle che dovranno essere utilizzati con la funzione Write( ) della "dos.library" (per il FileHandle dell'output standard è sufficiente una chiamata ad Output( ) della "dos.library").

Queste limitazioni, pur grosse, permettono però di scrivere programmi in C estremamemte compatti quanto a dimensioni (analoghi a quelli scritti in assembler!) e, soprattutto, di evitare imprevisti effetti collaterali, generati dal modulo di startup o dalle routine standard del linguaggio "C". Nell'eseguibile finale ci sarà solo quello che avremo scritto noi e nient'altro: programmare in C diventa, allora, come usare l'assembler, solo che risulta molto più facile, perché dei registri, dello stack, degli offset si preoccupa il compilatore, non noi (ovviamente un programma di questo genere non risulta affatto portabile verso altri computer). Personalmente, ritengo poco conveniente scrivere codice in assembler su Amiga, se non per piccole funzioni critiche sotto il profilo temporale o dell'occupazione di memoria: tutto quello che può essere fatto in assembler (quasi) può essere fatto in C, a patto di rinunciare all'ambiente standard e alle sue facilitazioni.

Torniamo alla struttura del programma: dopo il primo hunk, gli altri (il loro numero dipende da tanti fattori) possono contenere segmenti di codice o di dati. Se però si specifica l'opzione SMALLCODE (o SC) sulla linea di comando di blink, tutti i frammenti di codice presenti nel nostro sorgente (composto anche da più di un file) verrano radunati in un unico hunk di nome "text". Se si specifica l'opzione SMALLDATA (SD) tutti i dati, anche quelli non inizializzati, verrano radunati in un unico file di nome "__MERGED". Il nostro programma sarà composto così da

due soli hunk, caricati in due segmenti diversi di memoria. Occorre fare attenzione a una sola cosa: alcune funzioni della libreria standard del C possono generare un hunk vuoto, nonostante le opzioni di blink, portando a tre gli hunk effettivi; per sincerarsene basta consultare la mappa degli hunk prodotta da blink.

Il nostro programma permette di rendere residenti in memoria fino a tre hunk, creando una MemList con tre MemEntry che localizzano i segmenti del programma stesso (nulla vieta, comunque, di proteggere un numero più elevato di hunk, scegliendo una MemList con più MemEntry). Il metodo utilizzato prevede che tutto il programma resti residente in memoria, sopravvivendo al reset. E' la funzione PreparaMemList(mlptr, numentries) che individua la nostra SegList: ciò è realizzato trovando l'indirizzo del nostro processo con la funzione Exec FindTask(0) (offset $feda) e poi quello della struttura CommandLineInterface associata al processo (che deve dunque essere lanciato da CLI o dalla Shell e non da Workbench), attraverso il campo rt_CLI (moltiplicato per 4 con la macro BADDR( ), come tutti i BPTR). In tale struttura il BPTR cli_Module contiene esattamente l'indirizzo del nostro primo hunk. Rispetto a tale indirizzo, a un offset di -4 byte sta la lunghezza in byte dell'hunk, a un offset pari a 0 sta il BPTR al prossimo segmento, e a un offset di +4 sta l'inizio del codice utente. Questi valori, e quelli analoghi degli hunk seguenti, vengono usati dalla funzione per costruire la MemList da aggiungere a KickMemPtr. Se il numero delle MemEntry (parametro "numentries") risulta insufficiente (perché inferiore al numero di hunk) la funzione restituisce il valore 0, altrimenti l'indirizzo di cli_Module; ponendo a 0 il suo contenuto, il Dos non sarà più in grado di rimuovere la nostra memoria dal sistema quando il programma terminerà, ed avremo così ottenuto l'effetto desiderato.

Nome	Resident	Pri	Tipo	Vers	Flag
exec.library	$fc00b6	120	LIBRARY	33	$00
expansion.library	$fc516c	110	LIBRARY	33	$81
potgo.resource	$fe4880	100	RESOURCE	33	$01
keymap.resource	$fe4fe4	100	RESOURCE	33	$01
cia.resource	$fc450c	80	RESOURCE	33	$01
disk.resource	$fc4794	70	RESOURCE	33	$01
misc.resource	$fe4774	70	RESOURCE	32	$01
ramlib.library	$fe49cc	70	LIBRARY	33	$80
graphics.library	$fd0a3c	65	LIBRARY	33	$01
keyboard.device	$fe502e	60	DEVICE	33	$01
gameport.device	$fe507a	60	DEVICE	33	$01
timer.device	$fe90ec	50	DEVICE	33	$01
layers.library	$fe0d90	31	LIBRARY	33	$01
console.device	$fe510e	20	DEVICE	33	$01
trackdisk.device	$feb3dc	20	DEVICE	33	$01
intuition.library	$fe0378	10	LIBRARY	33	$81
alert.hook	$fc323a	5	UNKNOWN	33	$01
mathffp.library	$fe424c	0	LIBRARY	33	$81
workbench.task	$feb400	0	TASK	33	$00
dos.library	$ff425a	0	LIBRARY	33	$00
strap	$fe8884	60	UNKNOWN	33	$01

Tabella 1: Il valore di alcuni campi della struttura Resident dei moduli residenti presenti di default nella versione 1.2 (33) del Kickstart.

La funzione CercaResident(nome, vers, tagpp) ricerca prima nella lista ResModules, poi in KickTagPtr un modulo che abbia il nome e la versione precisati (se "vers" è pari a 0, non tiene conto del numero di versione) e ritorna una UWORD di flag che indica se e dove è stato trovato il modulo (si veda il listato per l'esatto significato dei suoi bit). Se il modulo esisteva già, il contenuto di "tagpp" sarà un puntatore alla corrispondente struttura Resident.

La funzione è utile per stabilire se il modulo è già residente o se si è già richiesto ad Exec di renderlo residente al prossimo reset; in tali casi il nostro programma termina senza duplicare il modulo. Se invece il programma si trova nella lista di ResModules, ma non in quella di KickTagPtr, significa che il vecchio modulo non potrà sopravvivere al reset e quindi è possibile aggiungerne un'altra copia (ciò avviene se un programma ha modificato KickTagPtr cancellando il vecchio romtag).

La funzione PreparaResident(tagptr, vers, pri, nome, id, init) pone nella struttura Resident indirizzata da "tagptr", la versione, la priorità, il nome, la stringa di identificazione e l'indirizzo della funzione passati come parametri; il modulo sarà inizializzato con il flag RTF_COLDSTART e il tipo NT_UNKNOWN.
La funzione AddResident(tagptr, mlptr, kickarray) collega il nostro programma a ExecBase, utilizzando la MemList il romtag e l'array specificati, e ne ricalcola KickCheckSum.

La funzione ResetFunz( ) contiene il codice che verrà chiamato da Exec durante il reset; avendo il programma scopi puramente dimostrativi, essa si limita a chiamare DisplayAlert( ) di Intuition con una stringa di avvertimento.
E' questa la funzione che dovrebbe essere modificata dal lettore per fare degli esperimenti. Si tenga presente, a tal proposito, che tutti i dati esterni del programma sono disponibili a questa funzione quando si carichi in A4 l'indirizzo dell'hunk dei dati: ciò si può ottenere precisando l'opzione -y sulla linea di comando di lc2, oppure, a partire dalla versione 5.0, con la parola chiave __saveds premessa al nome della funzione o ancora chiamando la funzione geta4( ) all'inzio di ResetFunz( ).

Si tenga presente, infine, che, essendo in corso un reset, tutti i valori che dipendono dal sistema, come SysBase e DOSBase, vanno ricalcolati, le librerie e i device riaperti, mentre il Dos non è ancora stato lanciato e dunque, se si vuole leggere un disco, per controllare, ad esempio, se contiene un virus, si deve farlo attraverso il "trackdisk.device".

Il Debugging

Regole fondamentali

Philip "Phil" Roberts

Qualsiasi imperfezione, difetto o mancanza che produce risultati erronei o inesatti viene chiamato BUG. Ed il DEBUGGING è l'arte della ricerca e della correzione dei bug e dei problemi ad essi relativi. Il debugging si avvale di tecniche scientifiche, ma in definitiva, quando tutto è stato detto e tutto è stato fatto, sono l'abilità e l'accortezza, o la mancanza di tali qualità, a determinare l'esito del debugging. Prendendo in considerazione tutte le fasi necessarie per lo sviluppo di un programma: progettazione, stesura, debugging e verifica, ci rendiamo conto che il debugging occupa un tempo maggiore di tutti gli altri stadi messi insieme. Questo articolo contiene regole e tecniche che potranno essere utili per il debugging.

Regole generali per il debugging

1. Non date mai nulla per scontato. Non è detto che un programma attualmente funzionante non contenga dei bug. Il fatto che un certo algoritmo abbia sempre funzionato correttamente non significa che esso vada bene per una successiva serie di condizioni e di dati. Che una particolare subroutine o un blocco di codice abbia funzionato per anni non vuol dire che sia privo di bug. Siamo sicuri di non avere mai apportato alcuna modifica?

2. Assicuratevi che il bug esista davvero. Abbiamo previsto correttamente ciò che doveva accadere? Il risultato che ci aspettiamo è quello esatto?

3. Controllate innanzitutto le cose più semplici e ovvie. Se ci sono parecchi elementi simili, indagateli in ordine crescente di complessità. Nei programmi per microcomputer, sono i messaggi d'errore che compaiono sullo schermo a dettare gli elementi da controllare per primi.

4. Siate sistematici e metodici in tutte le procedure. Utilizzate il buon senso in maniera organizzata.

5. Tenete degli appunti che registrino tutti i tentativi e le procedure utilizzate durante il debugging, ricordando in particolar modo tutti i casi verificati. Rileggendo il registro più avanti potremo sapere cosa provare e cosa evitare in caso avessimo dei problemi simili. La conoscenza di casi fastidiosi potrà essere utile per programmi che scriveremo in futuro.

6. Ricordate che ogni tentativo di correzione d'un errore può portare ad una moltitudine di nuovi errori di cui il programma originale era privo. Siate particolarmente cauti nel cambiare la logica di un programma. Potremmo iniziare con una istruzione condizionale "salta se uguale" come

```
IF A = B THEN PIPPO
```

Ci accorgiamo che non agisce correttamente, quindi proviamo

```
IF A <> B THEN PLUTO
```

Ma si presenta una nuova serie di errori, e quindi proviamo un contatore

```
IF A = 12 THEN PIPPO ELSE A= A + 1
```

Comunque non abbiamo inizializzato la variabile.

7. Correggiamo uno e un solo bug alla volta. Se apportiamo troppi cambiamenti simultaneamente, non sappiamo quale soluzione corregge quale bug e perché. Prendiamo il tempo che ci serve, cercando sempre di perfezionare e migliorare le nostra capacità di debugging così che la prossima soluzione necessaria sarà più evidente.

8. Il punto in cui l'errore viene rilevato (dove si mostra il bug) non è detto che corrisponda al punto d'origine del bug (dove si è creata la condizione d'errore) ma potrebbe essere in un altro punto.
L'istruzione DATA è un esempio calzante.
Considerate:

```
DATA 15,20,25,30,35
FOR I = 1 TO 10
READ A(I)
NEXT I


OUT OF DATA
```

Il nostro errore non si trova alla linea READ(I), ma piuttosto nella prima, dove si trovano soltanto cinque elementi data anziché i dieci necessari.

9. Ricordate che la sintassi di un programma AmigaBASIC non viene controllata mentre lo state digitando. Gli errori di sintassi non si mostrano sino a che non date RUN.

Tipi di bug

Ora che abbiamo stabilito qualche regola per il debugging, quali sono i tipi di bug che non aspettano altro se non di infestare i nostri programmi ? Prima di tutto, a differenza di altri linguaggi ad alto livello, non c'è nessuna gerarchia per i bug in BASIC. Non esistono errori d'allarme, o fatali. Ogni errore può passare totalmente inosservato, come quando si utilizza un algoritmo sbagliato senza saperlo, oppure l'esecuzione del programma termina con un messaggio d'errore o la macchina si "blocca".
I bug di battitura possono caratterizzarsi come minori, maggiori e catastrofici, con tutti i tipi di categorie intermedie. Il metodo qui utilizzato classifica i bug nelle classi: di sintassi, di logica, data, variabili, array, aritmetici, d'operazione logica, d'input/output, e di tipo misto.

Errori di sintassi

1. Errori tipografici come 0 al posto di O oppure 1 per I.
2. Errori di punteggiatura come lo scambio del punto e virgola con la virgola.
3. Virgolette o parentesi non usate in coppia.
4. Spazi bianchi usati impropriamente come nell'inserimento di data.
5. Parole riservate usate come nomi di variabili.
6. Riferimenti a label inesatti.

Errori logici (di flusso di controllo)

1. Salto a una label errata.
2. Mancata considerazione di tutte le condizioni possibili e di salto appropriato.
3. Scorretta esecuzione d'istruzioni che appaiono dopo l'istruzione di salto.
4. Impossibilità del programma di terminare.
5. Istruzioni IF annidate utilizzate impropriamente.
6. Cicli impropriamente inizializzati.
7. Cicli con indice improprio.
8. Impossibilità di un loop di concludersi.
9. Cicli eseguiti una volta in più o in meno del dovuto.
10. In AmigaBASIC tutti i cicli vengono eseguiti almento una volta. E' questo ciò che si desiderava utilizzando un ciclo anziché un salto condizionale?
11. Uscite da cicli scorrette.
12. Trasferimenti da altre parti del programma nel mezzo di un loop.
13. Richiamo di una subroutine errato.
14. Uscite da subroutine scorrette.

Errori di DATA

1. Formato dei dati improprio, come l'uso di mm/gg/aa per una data quando veniva richiesto gg/mm/aa oppure mm-gg-aa.
2. Tentativo di inserire troppi o troppo pochi valori.
3. Istruzioni DATA e READ male accoppiate.
4. Controllo del programma trasferito nelle istruzioni DATA e valori data eseguiti come istruzioni di programma.

Errori di variabile

1. Mancata corrispondenza della dichiarazione di variabile con il data.
2. Utilizzo di variabili non inizializzate.
3. Stessa variabile utilizzata in più d'un contesto.
4. Stesso nome di variabile digitato in maniere diverse.
5. Nomi di variabili simili utilizzati e interpretati dal computer come una stessa variabile.
6. Un contatore non propriamente inizializzato.
7. Una variabile d'indice che non assume l'intera gamma di valori ad essa propri.

Array e matrici

1. Alcuni array non propriamente DIMensionati.
2. Alcuni array non inizializzati a zero o con i valori appropriati.
3. Un indice non all'interno dei limiti.
4. Uso di un indice non-intero.
5. Il primo elemento dell'array viene contrassegnato con 0 o con 1? Corrisponde al valore dell'indice?
6. Un indice si riferisce all'elemento dell'array sbagliato.
7. L'ordine degli indici è stato scambiato come quando si intende C(1,2) ma si usa invece C(2,1).
8. Numero di indici scorretto.

Errori aritmetici

1. Sono stati tentati modi di computazione misti.
2. L'accuratezza delle conversioni del tipo "numero" è stata male interpretata.
3. Calcoli effettuati su stringhe.
4. Superati i limiti massimo o minimo.
5. Effettuarsi del troncamento o arrodondamento.
6. L'inaccuratezza della rappresentazione di un numero base 10 come numero base 2 non è stata presa in considerazione.
7. La gerarchia (ordine) delle operazioni è stata male interpretata e/o male utilizzata.
8. Tentativo di divisione per zero.
9. Utilizzo di forme indefinite come 0/0, o 0 alla potenza di 0.
10. Il dominio e il campo delle funzioni aritmetiche sono stati male interpretati, usando per esempio la radice quadrata di un numero negativo.
11. Uso di complicati metodi di calcolo scorretti.

Errori di operazione logica

1. Le operazioni Booleane NOT, AND, e OR non sono state usate correttamente, o la loro precedenza non è stata usata

propriamente.
2. L'operatore + viene usato come concatenazione con le stringhe. E' questo che si desiderava?
3. I confronti non sono stati manipolati corretamente. Non si è tenuto conto degli spazi bianchi e dell'allineamento della virgola decimale.
4. Valori numerici sono stati confrontati con valori di stringa.
5. Le relazioni di confronto non sono state manipolate correttamente. I termini "al massimo", "non maggiore di", "al minimo", e "non minore di" sono stati mal compresi.
6. Quale è la gerarchia tra <, AND, ecc.? è utilizzata propriamente, come in I<(A AND B) vs. (I<A) AND (I<B)?
7. Quale valore è assegnato al valore VERO? è -1 o 0 o 1? Per esempio in

```
IF A THEN GOTO PIPPO
```

Il salto al label PIPPO è eseguito quando A è -1 o per quale valore?

8. Come sono trattate le espressioni Booleane nelle istruzioni IF...THEN...ELSE ?
Per esempio nella linea

```
IF (I<>0) AND (J/I=5) THEN K = 10
```

il controllo passa alla linea successiva quando I è zero, o viene mostrato un messaggio di divisione per zero?

Errori di Input/Output

1. Lettere maiuscole o minuscole non usate appropriatamente.
2. Inserimento di valori inappropriati in risposta a richieste di INPUT, LINEINPUT o INKEY$.
3. Utilizzo improprio di porte input o output.
4. Codici di stampante usati scorrettamente.
5. Non tutti gli attributi di file corretti.
6. I file corretti sono stati aperti (OPEN) prima di essere usati e chiusi (CLOSE) appena usati?
7. Una condizione di end-of-file è stata usata scorrettamente.

Una lista di utili domande

1. Il programma può tollerare i casi eccezionali o soltanto quelli ovvi e di routine?
2. Si sta usando la funzione appropriata?
3. La funzione è mancante?
4. La sequenza casuale è sempre la stessa al momento di partenza?
5. Vengono controllati i valori di input? A che punto del programma?
6. Ci sono delle variabili che vengono considerate locali?
7. Si stanno usando i corretti algoritmi?
8. Il programma adempie i compiti per i quali era stato progettato, o ne porta a compimento altri ?

Facendo riferimento a questa lista, dovrebbe esservi più facile isolare la porzione o la funzione del programma che ha generato l'errore.

C'è poi un'altra lista che dovremmo tenere a fianco del nostro computer per ricordarne le limitazioni. Ovviamente essa varierà a seconda del modello del microcomputer.

Limitazione del Basic

1. Intervallo dei valori ammissibili per gli interi.
2. Intervallo dei valori ammissibili per i numeri a singola precisione.
3. Intervallo dei valori ammissibili per i numeri a precisione doppia.
4. Tipo di valore dei risultati di operazioni aritmetiche.
5. Lunghezza massima dei nomi di variabile.
6. Lunghezza massima di una stringa.
7. Lunghezza massima di una linea d'istruzioni.
8. Massima quantità di memoria RAM.
9. Massimo numero di parentesi consentito in una singola linea.
10. Massima grandezza dell'indice di un array.
11. Massimo numero di dimensioni in un array.
12. Massimo numero di subroutine annidate.
13. Massimo numero di cicli annidati.
14. Minimo numero di byte di memoria necessari al programma per le linee d'istruzioni (numeri di linea, puntatore alla prossima linea, e per fine linea), per parole riservate, nomi delle variabili, per operatori e caratteri.
15. Minimo numero di byte di memoria necessari durante l'esecuzione del programma per le variabili, i GOSUB, i cicli FOR...NEXT, e per i livelli di parentesi.

Ora che ci siamo fatti un'idea di ciò che può andare storto e di quali sono i limiti di AmigaBASIC, siete pronti ad apportare le modifiche necessarie per il perfetto funzionamento di un vostro programma.

Interchange File Format

Dentro il mondo dell' IFF

Sebastiano Vigna

Uno dei punti di forza di Amiga è senza dubbio quello di possedere un formato di scambio dei dati efficiente e diffuso. In ambiente MS-DOS ogni programma vive in un mondo isolato, a sé stante, e lo scambio di dati, se possibile, avviene con complicate operazioni di conversione. Su Amiga, un qualunque file disegnato con Deluxe Paint può essere caricato da un programma, poniamo, di animazione, per usarlo come sfondo. I suoni generati per sintesi elettronica da Synthia si possono usare con Music-X. Qualunque applicazione supportante il clipboard può tagliare ed incollare porzioni di testo o immagini provienienti da altri programmi. Evidentemente, tutto questo non avviene per caso o perché on a de la chance: dietro questo comportamento del software ci sono delle persone previdenti.

La prima scoperta, che può destare sorpresa, è che le persone in questione non sono tutte di casa Commodore. Tutti i nostri lettori penso avranno perlomeno sentito parlare qualche volta dello standard IFF (o ILBM, FTXT, 8SVX, SMUS...). Per "standard", in questa sede, intendo un insieme di regole molto precise che specificano come deve essere strutturato un certo tipo di dati. IFF significa semplicemente Interchange File Format (formato per lo scambio dei file), ma dietro questa sigla si nasconde un'elegante filosofia, un sofisticato design e, soprattutto, un grande numero di standard, in costante aumento.

L'idea alla base dell'IFF è che i programmi dovrebbero usare normalmente un formato di interscambio comune. Non bisogna fornire utility di conversione, ma uniformare il modo di registrare i dati. In altre parole: un po' di lavoro in più ad ogni Open e Save vale certo il non dover scrivere, mantenere ed utilizzare i programmi di conversione (che sono, e ben lo sa chi abbia mai provato a scriverne, quanto di più noioso e complicato esista in fatto di programmazione). L'utente si trova sgravato dal dover imparare l'uso di tali programmi, e chi scrive il software non ha che da guadagnarci.

La ragione per cui l'IFF ha avuto tanto successo e diffusione su Amiga nasce dal fatto che lo standard era fondamentalmente già pronto quando la macchina uscì: era stato infatti preparato dalla Electronic Arts non solo per Amiga, ma anche per il Macintosh, in cui giacciono le origini di alcune idee dell'IFF; in particolare, il fatto che gli identificatori usati all'interno di un file siano lunghi quattro caratteri è dovuto all'analogo espediente usato dalla Apple per marchiare formati del clipboard, risorse eccetera.
L'altra referenza (non poteva mancare) è la Xerox; non contenta di aver inventato pull-down menu, mouse, window, le stampanti laser, i linguaggi ad oggetti e non so più che altro, ha proposto uno standard di interscambio dati (XScript) dietro a cui stava un'idea geniale: un documento può essere formato da molte parti separate (testo, immagini, equazioni matematiche, note,...), e un editor dovrebbe permettere di alterare quelle che riconosce ma (ecco l'idea fondamentale) lasciare inalterate al loro posto le altre. Questa filosofia è migrata in parte nell'IFF, anche se purtroppo molti programmi non sono attenti a questo fatto, che è uno dei punti di forza dello standard.

Come avrete capito, lo scambio di immagini, così comune in ambiente Amiga, non è che la punta di un iceberg. In effetti, le immagini non sono che uno dei tanti gruppi FORM che l'IFF permette. In particolare, Amiga usa il FORM ILBM (Interleaved Bitmap), mentre su Mac è anche utilizzato il PICT (che contiene immagini in formato QuickDraw). Ci sono altri gruppi (PROP, LIST e CAT) che permettono di organizzare più efficacemente gruppi di dati omogenei, ma qui non entreremo nei dettagli.

Il nome ILBM deriva dal fatto che un'immagine raster (cioè formata da punti) di più bitplane viene memorizzata riga per riga, alternando i bitplane; in questo articolo ci soffermeremo in particolare su questo tipo di FORT IFF, e accenneremo solo brevemente agli altri. Una documentazione esauriente (per la precisione, LA DOCUMENTAZIONE) sull'argomento si può trovare nell'Amiga ROM Kernel Reference Manual: Includes & Autodocs, pubblicato dalla Addison-Wesley. Mi permetterò, qua e là, di mentire deliberatamente, al fine di non appesantire l'argomento con continui "ma esiste inoltre...", "sebbene si potrebbe anche..." eccetera. Ci sono diversi tipi di gruppi definiti nello standard IFF, ma quello che ci interessa qui sono i gruppi di tipo FORM, ove sono memorizzati dati. Questi possono essere a loro volta contenuti in strutture di tipo più complesso ed articolato, in pieno stile di programmazione ad oggetti, ma qui non ci occuperemo di questo aspetto.
Un gruppo FORM possiede sempre un identificatore di quattro caratteri, che ne specifica il contenuto. Gli identifi-

catori disponibili sono in continuo aumento, dato che vengono continuamente scritti nuovi programmi che hanno bisogno di memorizzare nuovi tipi di dati.
Il gruppo di identificatori originalmente concepiti comprende:

-FTXT: testo formattato, con font, dimensioni, etc. Tra i vari tipi è forse il meno diffuso (purtroppo!!). Excellence! lo usa normalmente.
-ILBM: ne abbiamo già parlato
-SMUS: è un formato che consente di memorizzare spartiti piuttosto semplici, ed è usato da SONIX
-8SVX: memorizza suoni campionati

Tanto per cultura, ecco alcuni degli acquisti più recenti; alcuni non sono ancora stati completamente ufficializzati dalla Commodore-Amiga, che funge da centro di controllo per l'inserimento di nuovi FORM nello standard:

-ANIM: animazioni; creato in origine per il VideoScape
-MIDI: contenuto ovvio, ma la struttura non è stata ancora completamente specificata
-PGTB: creato da John Toebes, all'interno del Lattice C, come formato diagnostico del crash di un programma
-WORD: formato creato dalla New Horizons per ProWrite; simile al FTXT, ma con più tipi di dati
-C100: creato in Italia dalla Cloanto per il suo WP

Potrei nominarne molti altri, ma la cosa importante è che pressoché tutti i produttori di software per Amiga usano IFF. Questo non significa che tutti i programmi possano scambiare i dati. Per esempio, ProWrite usa il FORM WORD, mentre Excellence! usa il FORM FTXT. E' però probabile che in un prossimo futuro i due programmi possano leggere l'uno i file dell'altro. E in ogni caso, il design complessivo dei due tipi di dati è così simile da rendere possibile a chiunque conosca un minimo di programmazione la scrittura di un programmino di conversione.

Tanto per chiarire la situazione, che potrebbe a questo punto sembrare un po' nebulosa, diciamo che un file IFF è composto da un FORM: vale a dire, comincia esattamente con i quattro caratteri ASCII 'F', 'O', 'R' ed 'M', come potete verificare facilmente battendo da CLI "Type <nomefile> hex", ove <nomefile> è il nome di un file IFF. Immediatamente dopo (come avviene per tutti i gruppi IFF) viene una parola lunga (quattro byte) che specifica la lunghezza totale del FORM; nel caso questa sia dispari, un byte di allineamento viene aggiunto alla fine, ma non viene contato.

La parola lunga è seguita da un identificatore di quattro lettere, che può essere, come abbiamo detto, ILBM, SMUS, FTXT eccetera. Noi ci occuperemo solo di file ILBM. Notate che la preferenza data alle parole lunghe per gli identificatori è in fondo una conseguenza del fatto che i registri di un 68000 sono a 32 bit.
A questo punto comincia il contenuto del gruppo FORM. Esso è formato da più parti, dette chunk, marchiate ciascuna con un identificatore di quattro caratteri seguito dalla lunghezza del chunk stesso (vale sempre la regola dell'allineamento, come per i gruppi). Alcuni chunk sono obbligatori, altri opzionali. La grande forza dell'IFF sta nel fatto che fondamentalmente noi possiamo dimenticarci completamente di quelli che non capiamo.

In particolare, tre chunk sono obbligatori: BMHD, CMAP e BODY.
Il primo contiene una struttura di tipo BitMapHeader, che specifica il formato dell'immagine, ovvero le dimensioni e il numero di colori. Il secondo enumera le componenti RGB di ciascun colore della palette. Sono disponibili 8 bit per ogni componente, il che ci lascia specificare circa 16.000.000 (2^24) colori. Nel caso una macchina usi solo una parte degli otto bit (per esempio, su Amiga abbiamo 4 bit per componente) questa deve venire spostata fino in cima al byte: in pratica, quando leggeremo un file IFF dovremo muovere a destra ogni byte di quattro bit.

L'ultimo chunk, BODY, è il più importante nel senso che contiene i dati veri e propri. Il formato è intercalato, in quanto non memorizziamo ogni bitplane di fila, bensì una riga per ogni bitplane. Se il procedimento è piuttosto inefficiente dal punto di vista della compressione, è però molto flessibile: un programma che necessiti di solo una parte dell'immagine, in particolare, si potrà fermare alla riga che gli serve, senza leggere tutto il file.

Due particolari vanno aggiunti: è possibile che sia presente un bitplane "fasullo", detto di masking, che può servire a definire una "maschera" attraverso cui va pensata l'immagine (ad esempio, se si tratta di un pennello ritagliato in modo irregolare, il masking può definire la forma originale con cui è stato definito). Inoltre, ogni riga può essere compressa con un algoritmo specificato nel chunk BMHD: attualmente è in uso solo l'algoritmo ByteRun1, di cui parleremo in seguito. Infine, un chunk opzionale CAMG (Commodore-Amiga) può contenere la parola corrispondente ai registri video dell'hardware. In questo modo, è possibile informare chi legge il file del fatto che l'immagine è in HAM o Extra_Halfbrite, in alta o bassa risoluzione eccetera.

E' possibile incontrare molti altri chunk. Per esempio, CRNG specifica i range di colori del DeLuxePaint e AUTH il nome dell'autore. Un'applicazione, quando legge un file, deve utilizzare i chunk che conosce e non badare agli altri. In questo modo non si rinuncia alla possibilità che un programma "arricchisca" un file di informazioni proprie, e contemporaneamente si ha una grande compatibilità.

Attualmente, non esiste alcuna funzione, nelle librerie di Amiga, che permetta di leggere un file IFF; dovremo quindi arrangiarci da soli. Commenterò ora un programma scritto in C a questo scopo. Per principio, è carente in fatto di controlli: lanciato su un file IFF mal scritto potrebbe anche condurre al crash, ma l'inserimento di numerosi controlli avrebbe reso incomprensibile l'insieme.

Prima però del commento vero e proprio, due parole sull'algoritmo di compressione ByteRun1. Il formato è estremamente semplice:

- se leggi un numero tra -1 e -127, duplica il byte successivo per -n+1 volte
- se leggi un numero tra 0 e 127, leggi letteralmente gli n+1 byte seguenti
- se leggi -128, non fare nulla e passa al successivo (questo caso serve solo per l'allineamento)

Come si può facilmente vedere, il compattamento e lo scompattamento sono facili e veloci, ma le immagini, avendo spesso delle parti uniformi o vuote, vengono ridotte notevolmente; la cosa non è più vera nel caso l'immagine sia "rumorosa" in senso informatico, per esempio in modo HAM. La prima cosa da fare è aprire il file e controllare che i primi quattro byte siano costituiti dalle lettere "FORM". A queste seguono la lunghezza e il tipo, che deve essere "ILBM". Se tutto va bene, il ciclo successivo legge uno dopo l'altro i chunk del file. Nel caso il chunk non venga riconosciuto, viene semplicemente saltato utilizzando l'informazione sulla sua lunghezza, cioè, come già detto, la parola lunga immediatamente successiva all'identificatore. Attenzione: dato che se la lunghezza del chunk è dispari, esso viene allineato con un byte che non rientra nel conto, esso va saltato dopo una lettura o una scansione, altrimenti ci si ritrova nel posto sbagliato. A questo bada il test "if (l%2)".

Il programma riconosce solo i chunk BMHD, CMAP e CAMG, ma con questi si può leggere il 99% dei file IFF. Tra le cose che non prendiamo in considerazione ci sono i cicli di colori del DPaint, il chunk che contiene il nome dell'autore eccetera.
Dalla struttura BitMapHeader, che forma il contenuto del chunk BHMD, prendiamo in prestito l'ampiezza, l'altezza e la profondità dell'immagine (cioè il numero di bitplane), nonché il tipo di compressione usata e un flag che indica la presenza del masking. La struttura completa è all'inizio del programma. Dal chunk CMAP prendiamo invece, elencate sotto forma di byte, le componenti R,G,B per ogni colore della palette. Il chunk CAMG (Commodore Amiga) contiene infine una parola lunga (cioè 4 byte) la cui metà superiore è tuttora riservata. Quella inferiore, invece, contiene eventuali flag da settare nel campo NewScreen->DisplayFlags.

Il ciclo while più esterno viene ripetuto fino a che non si arriva ad un chunk BODY: dato che lo standard IFF assicura che questo è necessariamente l'ultimo, prima di lui abbiamo certamente trovato BMHD, CMAP e CAMG.

Prima di tutto si apre uno schermo sulla base delle dimensioni dell'immagine. Attenzione: possono capitare dei pennelli (sono anche lori immagini ILBM!) così piccoli da mandare in crash Amiga perché lo schermo che viene aperto è troppo piccolo per contenere i gadget di sistema (l'immagine potrebbe anche essere troppo grande, ma questa è già un'eventualità più rara).

Quindi bisogna modificare i colori dello schermo con la funzione SetRGB4 sulla base della CMAP. Per sapere quanti sono, è molto meglio affidarsi alla lunghezza della CMAP diviso tre anziché calcolare 2 elevato al numero di bitplane. Per esempio, il modo HAM ha una palette di 16 colori, ma 6 bitplane.

A questo punto viene l'algoritmo di scompattamento. Anche in questo caso ho fatto assunzioni "buone", ad esempio che non ci siano codici di compattazione che fanno debordare oltre una riga (fenomeno detto compressione "attraverso" le righe): lo standard IFF proibisce esplicitamente di comprimere da una riga all'altra.

L'algoritmo è così semplice da essere autoesplicativo; si noti solo che le if sono necessarie per evitare che mentre si sta decompattando il bitplane "fasullo" di masking questo venga scritto effettivamente.
Ho usato le funzioni ANSI memset e memcpy perché lo standard ANSI per il C incoraggia esplicitamente l'implementazione di queste funzioni in-line anziché tramite librerie, e se si usa un compilatore che rispetta questo consiglio (leggi Lattice C 5.0x) si ha un notevole miglioramento rispetto ad un ciclo scritto esplicitamente.

Una volta visualizzata l'immagine, aspettiamo per 10 secondi con la funzione AmigaDOS Delay( ) e poi usciamo. Se volete, potete mettere qualcosa di più sofisticato come un attesa dalla tastiera o dal mouse. Ho usato funzioni AmigaDOS anziché C per leggere il file perché il formato è identico, ma agganciandosi direttamente si va più in fretta. Non dovrebbero esserci problemi di compatibilità Manx/Lattice, posto che i possessori del primo eliminino gli #include <proto/...>.

Chi a questo punto vuole approfondire può riferirsi al testo citato all'inizio. Tenete però presente che la situazione è in continuo movimento, e le decisioni da prendere conducono a volte a scontri piuttosto pesanti. Ad esempio, di recente la NewTek ha sviluppato un sistema per visualizzare 4096 colori in alta risoluzione, modificando in tempo reale la palette durante il passaggio del pennello elettronico. Purtroppo, queste immagini vengono salvate come se niente fosse in un FORM ILBM, ed un programma che non è a conoscenza di questa tecnica visualizzerà solo spazzatura, saltando il chunk speciale della NewTek. La cosa ha causato non pochi problemi e discussioni, ed evidentemente è contro l'etica IFF.

Non volendo concludere su una nota negativa, vi informo del fatto che circolano voci su una possibile creazione da parte Commodore di una IFF.library che permetterebbe senza fatica di fare il parsing dei file IFF più contorti... quindi incrociate le dita!

caricato insieme all'utility si deve trovare nella stessa directory di Macro Keys e deve chiamarsi default.macros; modificandolo sarà possibile cambiare le macro di default disponibili subito dopo il caricamento dell'utility.
L'opzione HELP fornisce all'utente una serie di brevi istruzioni per l'uso dell'utility.
Per uscire dal programma è sufficiente cliccare sul gadget di chiusura della finestra di Macro Keys.

Snap!Plus

Spesso accade di voler catturare e memorizzare lo schermo di un qualsiasi programma durante il suo funzionamento.
In questi casi con un programma come Snap!Plus è possibile sbizzarrirsi nell'immortalare in qualsiasi momento lo schermo di un qualsiasi programma compatibile con il multitasking.
Snap!Plus, una volta eseguito, si mette in stato di attesa ed entra in azione ogniqualvolta l'utente prema una delle particolari combinazioni di tasti riconosciuta dall'utility:

- Amiga Sinistro-M permette di portare in primo piano lo schermo di Intuition presente sotto quello attualmente visibile.

- Amiga Sinistro-S fa apparire una finestra nella quale è possibile scrivere il nome del file sul quale memorizzare l'immagine presente sullo schermo (esclusi sprite e puntatori vari).
Se la memoria disponibile è sufficiente, Snap!Plus farà una copia momentanea dello schermo prima di salvarlo per congelare ogni movimento; altrimenti eseguirà una copia in tempo reale dello schermo.
Se il salvataggio ha successo, lo schermo lampeggia per un attimo, altrimenti significa che si è verificato un errore (mancanza di spazio sul device di memorizzazione, path errato, memoria insufficiente): ricordate di specificare sempre il path completo del file nel quale memorizzare l'immagine.

- Amiga Sinistro-D ha lo stesso effetto della combinazione precedente, con l'unica differenza che l'immagine viene congelata alcuni secondi dopo la pressione dei tasti, permettendo, ad esempio, di aprire dei menu.

- Amiga Sinistro-CTRL-C ha l'effetto di terminare l'esecuzione dell'utility e di rilasciare la memoria che occupava in precedenza.

Dmouse

Dmouse è un'utility che permette, in un colpo solo, di avere a disposizione:

- Un efficace blanker di schermo e di puntatore dopo un certo numero di secondi (i due valori in secondi sono programmabili mediante i parametri che Dmouse accetta se chiamato da CLI (per maggiori informazioni leggere il file doc su disco). I valori di default sono di 5 secondi per il mouse e di 300 secondi per lo schermo.

- Una utility che attiva automaticamente la finestra presente sotto al puntatore, o, quando si digita qualcosa alla tastiera, la finestra dove immettere il testo.

- Un acceleratore di mouse molto pratico che permette di muoversi con grande precisione o velocità a seconda delle esigienze; la velocità reale del mouse viene elevata ad una potenza ennesima con n definibile dall'utente (default 3).

- Una utility che rimpiazza un tasto con un comando CLI (vedere file doc nel disco).

- La possibilità di portare in primo piano una finestra semplicemente cliccando con il tasto sinistro nella parte visibile di essa, il numero di click necessari a portare la finestra in primo piano è programmabile.

- Sempre con il puntatore su di una finestra, tenendo premuto il tasto sinistro del mouse e premendo il destro, è possibile portare la finestra presente sotto il puntatore dietro tutte le altre.

- Con la stessa procedura descritta sopra, ma ponendo il puntatore al di fuori di ogni finestra, è possibile portare in primo piano lo schermo sottostante quello attuale! (ciò permette di spostarsi facilmente da uno schermo all'altro).

Particolari molto utili: non ha bisogno di una finestra CLI, e può essere perciò richiamato dalla startup-

sequence. Non è necessario richiamarlo con RUN.

IMPORTANTE: DMouse ha associato un particolare file DMouse-handler che deve essere registrato o nella directory L o nella directory dalla quale viene richiamato DMouse.

Screen-Shift

Screen-Shift è un'utility che permette di spostare la posizione dello schermo attivo all'interno dello schermo del monitor o del televisore, in maniera simile alla funzione delle preferences, e si rivela molto utile per cambiamenti temporanei per effettuare i quali sarebbe troppo lungo e poco pratico caricare tutti i 40 e passa KBytes delle preferences, con il vantaggio che solo una piccola parte dello schermo viene coperta dalla finestra di Screen-Shift, permettendo di vedere in tempo reale se l'immagine che ci interessa è centrata.
In altri casi, quando, ad esempio, si desidera vedere anche le parti nascoste di un' immagine in Overscan, Screen-Shift diventa l'unica risorsa disponibile, perché le preferences, mentre spostano lo schermo, lasciano vedere solo il pannello di controllo, nascondendo l'immagine che ci interessa.
Una volta richiamato, Screen-Shift presenta una finestra con nel centro un piccolo rettangolino; cliccando su di esso e spostando il mouse (tenendo premuto il tasto sinistro), lo schermo ne segue i movimenti.

KeyLock

Avete degli amici burloni, dei fratellini minori, dei figli o degli animali domestici che non aspettano altro che vi assentiate un attimo dalla vostra postazione di lavoro per smanettare (o camminare nel caso del gatto domestico) sulla tastiera del vostro Amiga e non sapete come difendervi da questi pericolosi ed a volte letali attacchi? KeyLock è l'utility che fa per voi.
Una volta richiamato, questo programma congela tutti gli input di sistema, disabilita la tastiera e il mouse e rende impossibile qualsiasi manovra eccetto lo spegnimento mediante l'interruttore principale finché non viene digitata da tastiera una ben determinata parola d'ordine (max 30 caratteri), modificabile agendo direttamente sull'eseguibile mediante un editor di file (istruzioni dettagliate sul file doc nel disco).
La parola chiave settata inizialmente nel programma è PASSWORD.

• Scherzi

Avvertenze per l'uso dei programmi-scherzo

I cinque programmi presenti nel drawer scherzi del disco di questo mese, sono programmi che, solitamente, girano in background, impegnando quindi parte del tempo macchina messo a disposizione dal microprocessore e spesso utilizzando senza troppe precauzioni l'hardware di Amiga. Può quindi accadere, facendone girare più di uno alla volta, che il sistema possa andare in crash.
Se dovesse succedere, limitatevi ad utilizzare solo uno scherzo alla volta e non avrete problemi (anche se a volte, ma più raramente, può accadere che uno scherzo entri in conflitto anche con un normale programma, presente negli altri drawer).
I programmi-scherzo presenti su questo numero di Amiga Magazine sono: Viacom, Sand, Snake, Blocks, Wavebench.

Viacom

Viacom genera sullo schermo del WorkBench una sorta di disturbo che assomiglia a quello presente nelle trasmissioni televisive con una cattiva ricezione. Un effetto molto difficile da esprimere a parole, ma molto efficace da vedere.

Sand

Sand più che uno scherzo è una piccola demo, molto simpatica, nella quale tre contenitori contengono tanti granelli di sabbia che seguono il puntatore del mouse.
La cosa interessante consiste nel fatto che i granelli di sabbia si fermano se incontrano ostacoli e tornano sempre nel loro contenitore grazie a tre filtri presenti sopra i tre contenitori. Per uscire cliccate in alto a sinistra (c'è un gadget di chiusura nascosto da qualche parte).

Snake

Snake assomiglia, per certi aspetti, alle tipiche demo delle linee che si muovono rimbalzando all'interno di una finestra, con una variante: la linea che si muove è in realtà una curva (ottenuta mediante il metodo di costruzione delle linee "spline") che si contorce e si muove sinuosamente creando un effetto divertente e visivamente interessante.
La finestra può essere dimensionata a piacimento e chiusa con il solito gadget.

Blocks

Blocks crea un effetto visivo molto piacevole grazie all'utilizzo del modo HAM interlacciato.
Il programma fa muovere in uno schermo nero una serie di quadrati che, mano a mano che si spostano, assumono tutte le infinite (o quasi) sfumature offerte da Amiga.

WaveBench

Infine, WaveBench fa ondeggiare lo schermo del Workbench creando un effetto molto fastidioso che ben presto fa sorgere nell'utente i tipici sintomi del mal di mare, rendendo impossibile (o quasi) l'utilizzo del Workbench.

Espandere un 500

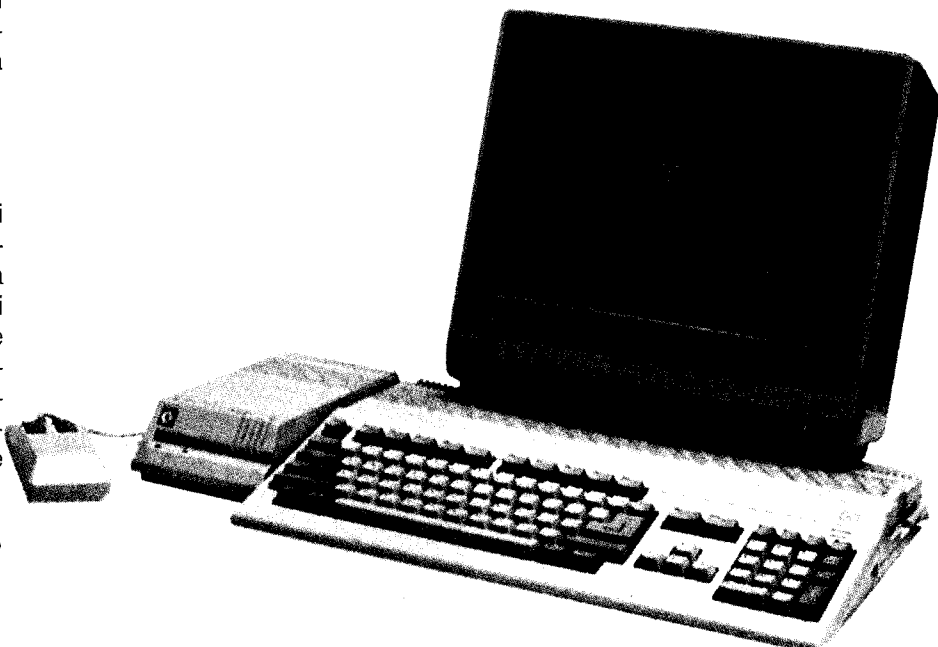
Arlan Levitan

Amiga 500 è ampiamente riconosciuto come uno dei computer migliori per il rapporto prezzo-prestazioni, visto che offre lo stesso impatto in termini di suono e grafica dei modelli superiori 1000 e 2000. Ma parlatene a un rivenditore che sappia distinguerli l'uno dall'altro, ed egli vi indicherà dove sta a suo parere la vera differenza tra le tre macchine: sicuramente nella superiore espandibilità di Amiga 2000. Certe differenze tra le tre macchine sono incontestabili, ma le possibilità di espansione di Amiga 500 sono più di quanto si pensi di solito. Poiché le porte parallele, seriale e video del 500 sono identiche a quelle dei fratelli maggiori, per esempio, non stupisce che quasi tutte le stampanti, i monitor e i modem supportati da 1000 e 2000 funzionino anche sul 500. Può essere sorprendente constatare la grande disponibilità di espansioni di memoria e dispositivi di memorizzazione per il piccolo della famiglia Amiga.

Più potenza

Prima di parlare di espansioni, si impone una considerazione di ordine generale: l'alimentatore di Amiga 500 non può sopportare molto più di un'espansione A501 e un drive supplementare. E' per questa ragione che quasi tutti i box di espansione per Amiga 500 hanno alimentazione propria. Se proprio si vuole superare il Mbyte di RAM, o due drive per floppy non sono sufficienti, bisogna pensare alla possibilità di sollevare l'alimentatore da un pesante carico di lavoro: per esempio ricorrendo ad alimentatori alternativi come quello della Phoenix Electronics, che per pochi soldi consente di aggiungere memoria e qualunque altro tipo di espansione, virtualmente, a piacere, proteggendo oltretutto il sistema dai picchi di tensione.

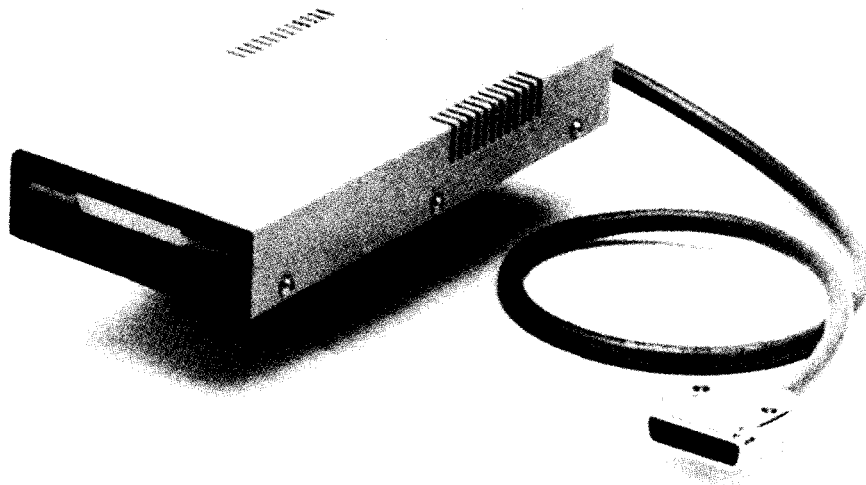
Più memoria

La maggior parte dei giochi per Amiga sono più che soddisfatti dai 512 Kbyte di RAM standard, ma i programmi più recenti, come Dungeon Master e Space Ace, richiedono un Mbyte di memoria. Per non parlare dei pacchetti ad alto impiego di grafica, per desktop publishing, animazione, CAD e via dicendo, per i quali un Mbyte di RAM è già poco e qualunque espansione è sempre bene accetta.
Il sistema operativo di Amiga dispone per giunta di un comodo ramdisk, utilizzabile per velocizzare le operazioni più ripetitive; peccato che lo spazio occupato dal ramdisk sia inutilizzabile dai programmi. In questo caso, aggiungere 512 Kbyte supplementari significa godere delle comodità del ramdisk e avere ancora la possibilità di far girare i programmi più usati.
Molti utenti, poi, espandono la memoria per trarre maggiore vantaggio dalle capacità di multitasking di Amiga. Naturalmente, più processi di elaborazione sono attivi

L'Hard disk Commodore A590 da 20 Megabyte

Il floppy disk Master 3-A della Oceanic America

simultaneamente, maggiore è la richiesta di memoria. Una soluzione economica e sicura è l'espansione Commodore A501, che porta la RAM totale di Amiga 500 a un Mbyte con la semplice introduzione della cartuccia nel box di espansione. Esistono espansioni di memoria realizzate da molti altri costruttori, ma l'ascesa del prezzo delle RAM le ha rese non sempre competitive nei confronti dell'espansione Commodore, che ha dalla sua il duplice vantaggio di avere un involucro protettivo (la maggioranza dei concorrenti vende semplici schede) e un calendario/orologio alimentato a batteria. Insomma, una comodità in più.

Dentro e fuori

Le espansioni di memoria per Amiga 500 che superano il Mbyte di RAM si dividono in due categorie, interne o esterne, secondo il tipo di installazione dentro o fuori da Amiga. Il primo ha costi più convenienti, ma comporta operazioni spiacevoli come aprire il computer e inserire una board aggiuntiva su quella vecchia. Simili espansioni sono prodotte, per esempio, da Progressive Peripherals e Spirit Technology. Della prima, l'EXP-1000 e l'EXP-8000 aggiungono rispettivamente uno e otto Mbyte. La SIN 500-2 della Spirit va, invece, a incrementi di 512 Kbyte, fino a raggiungere i due Mbyte. Comunque, se non si è esperti, è meglio lasciare eseguire l'operazione di installazione a un rivenditore qualificato. Per quanto riguarda le espansioni esterne, la Starboard2/500 si collega al bus di espansione del 500 e arriva a una capacità massima di due Mbyte. Gli interessati dovrebbero ricordarsi che questa scheda è incompatibile con l'A501 Commodore. Del resto, nel mondo delle espansioni per Amiga, non è infrequente che due prodotti di case diverse parlino linguaggi differenti. Due buoni consigli: assicuratevi molto bene che ciò che state per comprare sia perfettamente compatibile con la vostra configurazione, e possibilmente comprate dispositivi che includano un'altra porta identica al bus di Amiga, per poter concatenare le vostre espansioni.

Le espansioni da grandi

Il bus di espansione del 500 può anche essere utilizzato per aggiungere la possibilità di collegamento delle schede progettate per Amiga 2000. Queste comprendono, tra l'altro, espansioni RAM piuttosto convenienti. Il Toolbox di Expansion Technology e il SubSystem 500 della Pacific Peripherals ne sono esempi, in grado di alloggiare due schede standard per A 2000. In particolare, il SubSystem 500 offre un drive da 3.5" supplementare.
Comp-U-Save vende tre box di espansione, da due, tre e sei slot. Quest'ultimo viene venduto in kit e fornisce tre slot Amiga e tre IBM.

Espansioni di massa

Gli Amigofili che fanno ampio uso del 500 sentono sicuramente la necessità del doppio drive. Il Commodore 1010 è un'ottima unità per questa esigenza, ma un po' costosa. Case tipo California Access, Flexible Data, Megatronics, Oceanic America e Future Systems hanno in catalogo drive compatibili più economici con un cavo di interfacciamento più lungo (che non è un lusso inutile), dimensioni ridotte e circuiteria che, per esempio, elimina il fastidioso click-click emesso dal 1010 e da altri prodotti quando non vi sono dischi nel drive. Per lo stesso prezzo del 1010, Flexible Data propone l'FData-20, con due drive e alimentazione separata.

Quando il gioco si fa duro...

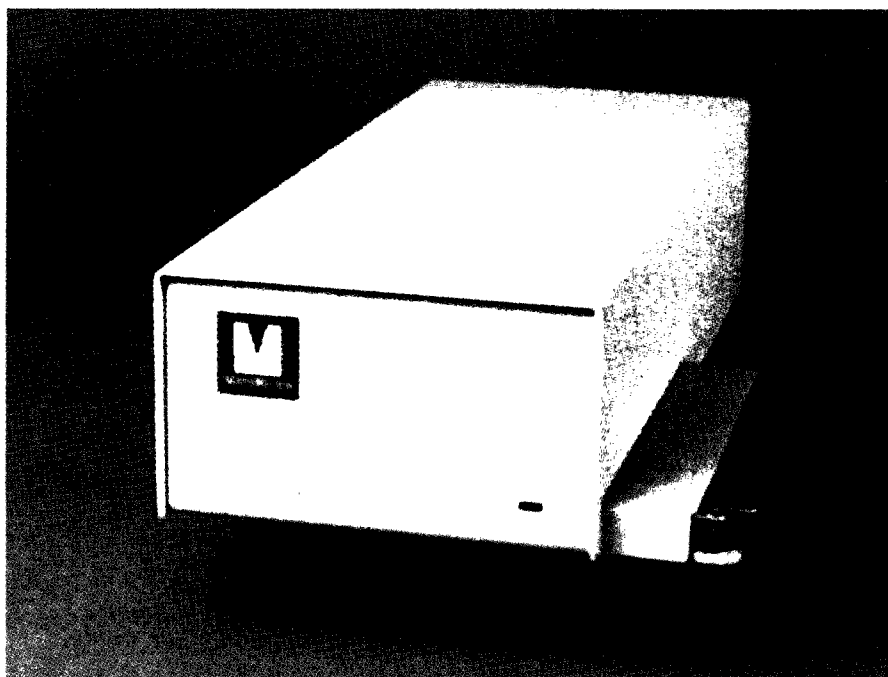
Ad Amiga 500 si possono anche collegare hard disk, capaci di velocizzare e rendere più comodo il lavoro. Basta pensare che un modello da 20 Mb contiene l'equivalente di più di 22 dischi. I prezzi partono da un milione dell'unità da 20 Mb, della Progressive Peripherals, per aumentare secondo la dimensione dell'hard. Tra i prodotti interessanti vi sono quelli della Supra (da 20 a 120 Mb), della GVP (40 e 80 Mb, basati su meccanica ultraveloce

Quantum, con RAM opzionale fino a due Mb) e della Escort (32 e 48 Mb). Gli hard disk compatibili con il Workbench 1.3 permettono di far partire il sistema direttamente dal disco rigido, e avvantaggiarsi delle prestazioni del Fast File System 1.3. La Commodore offre da circa sei mesi un hard disk: l'A590, da 20 Mbyte, si collega direttamente al bus di espansione, ma contiene una porta SCSI, a cui si possono collegare, per esempio, CD-ROM e altri hard disk.

500... all'ora

Non abbiamo notizia di schede acceleratrici per Amiga 500 basate su CPU Motorola 68020 o 68030, ma possiamo citare il Processor Accelerator della Creative Microsystem, una scheda aggiuntiva che raddoppia il clock del 68000 contenuto all'interno del computer portandolo da 7.16 a 14.32 MHz. Uno zoccolo ospita opzionalmente un coprocessore matematico 68881 con clock da 12.5 o 16 MHz.

Tutt'altra musica

Perché non installare altoparlanti aggiuntivi sul vostro 500? Costano poco, vengono da costruttori più che affidabili (come gli Yamaha DM-01) e possono spalancare le vostre orecchie dalla meraviglia, di fronte alle capacità sonore di Amiga. Se poi avete talento musicale potete dotarvi di un sistema ad hoc completo di interfaccia MIDI, tastiera e software opportuno. Pensateci...

Espandibile!

Concludendo, in fatto di espansioni Amiga 500 non sfigura nei confronti di alcun concorrente. Può non contenere un numero altissimo di slot vuoti sebbene ugualmente ingombranti, ma può soddisfare le esigenze di chiunque con configurazioni limitate solamente dalla fantasia, dal budget del proprietario e dallo spazio sulla scrivania. ▲

Unità di espansione da due Megabyte della Microbotics' Starboard 2/500

Gli Sprite

Parte da questo mese una nuova rubrica dedicata a tutti gli utenti di Amiga che da poco si sono avvicinati alla programmazione.

Gianni Biagini

Sicuramente ogni possessore di Amiga ha avuto modo di osservare la fluidità di movimento dei più svariati oggetti che compaiono nei videogame e sicuramente avrà notato come il puntatore del mouse, pur spostandosi, non cancelli gli schermi sottostanti.
La gestione delle pagine grafiche di Amiga è divisa in due componenti fondamentali: i PLAYFIELD e gli SPRITE che rappresentano più o meno gli sfondi e gli oggetti in movimento. Principali caratteristiche degli sprite sono: velocità di spostamento (la loro gestione avviene in DMA ossia Direct Memory Access che è una sorta di interruzione che obbliga il 68000 a cedere il BUS liberando così il microprocessore dai lavori di trasferimento dati), priorità relativa tra loro (esiste un particolare priority resolver che permette di visualizare solo le parti "non coperte" da oggetti di priorità superiore), qualsivoglia altezza e possibilità di utilizzare, per gli sprite "semplici" quattro colori di cui uno trasparente. Vediamo ora un po' più a fondo come definire l'immagine di uno sprite: si è scelto al riguardo l'uso di due dei più conosciuti linguaggi di Amiga, il Basic e il C.
Premettiamo che la trattazione sarà differente poiché ogni utente Amiga ha a disposizione, nel volume EXTRAS, uno sprite editor che crea file facilmente leggibili dal comando OBJECT.SHAPE in Basic, ma che non hanno senso in ambiente C.
Interessiamoci ora alla creazione di uno SHAPE ossia dell'immagine di uno sprite e vediamo come codificarlo.
Ogni sprite è formato da una serie sequenziale di WORD (configurazioni binarie a 16 bit) che a seconda della posizione in cui si trovano vengono interpretate come parametri o come linee di dato.
Come è facile intuire, per avere 4 colori è necessario creare uno schermo con 2 bitplane (questo perché un punto può essere solo acceso o spento e quindi un bitplane unico darebbe solo due possibili colori), mentre, se combinato con altri, ottiene una risoluzione cromatica espressa dalla formuletta: numero di colori=2^ numero di bitplane, per esempio, avendo 4 bitplane 2^4=16 colori. Poiché gli sprite semplici in Amiga hanno fino a quattro colori, affinché siano definiti servono 2 bitplane (ricordo che 2^2=4) e quindi non basteranno 16 bit per definire una linea dello sprite, ma ne serviranno 32. Niente paura, se non si può passare un dato a 32 bit poiché non ci starebbe in una singola locazione in memoria, spezziamolo pure in due parti, Amiga non si arrabbia per queste piccolezze. Ricapitoliamo: dobbiamo creare uno sprite, dobbiamo assegnare 4 colori e quindi considerare 2 bitplane e dobbiamo scrivere la sua shape in gruppi contigui di "parole" da 16 bit; siamo già sulla buona strada! Guardiamo attentamente la tabella 1: ci spiega chiaramente

bit 1	bit 2	colore
0	0	0
0	1	1
1	0	2
1	1	3

Tabella 1

come assegnare i colori voluti attraverso una serie di due bit, è esattamente quello che volevamo. Pensiamo adesso a due WORD (d'ora in poi utilizzeremo il termine word per indicare una configurazione di 16 bit) che diano una linea di 16 bit colorata così:

0123012301230123

il procedimento è semplicissimo: prendete carta e penna e risalite alla tabellina di prima:

parola 1 0011001100110011

parola 2 0101010101010101

guardate i bit corrispondenti delle due word appena scritte: identificano esattamente i colori della parola cercata, adesso siamo molto vicini al procedimento finale. Purtroppo i linguaggi di alto livello lavorano difficilmente su configurazioni binarie, perciò noi umani dobbiamo venirgli ancora incontro, basta osservare la tabella che ora vi presento:

configurazioni binarie	esadecimali
0011 0011 0011 0011	3333
0101 0101 0101 0101	5555

Cosa abbiamo fatto? Abbiamo sostituito ad ogni configurazione binaria di quattro bit il corrispondente valore esadecimale così il numero in base 2 0011 è diventato 3 e il numero 0101 è diventato 5. In base a che cosa ho trovato questi valori? Guardate la tabella 2 ci darà una mano: in essa è contenuto il codice di conversione da configurazione binaria a quattro bit a singola cifra esadecimale (che è quella che ci interessa maggiormente).
A questo punto forse conviene rivedere velocemente che passi si devono compiere per definire bene uno sprite:

- Prendere un foglio di carta quadrettata e una penna
- Disegnare un'immagine di 16 quadretti di larghezza
- Assegnare ad ogni quadretto un numero associato al colore e rappresentarne su due linee il codice binario come spiegato in tabella 1
- Dividere le configurazioni binarie ottenute in gruppi di quattro cifre
- Convertirle secondo la tabella 2 in codici esadecimali

Binario	Esadecimale
0000	0
0001	1
0010	2
0011	3
0100	4
0101	5
0110	6
0111	7
1000	8
1001	9
1010	A
1011	B
1100	C
1101	D
1110	E
1111	F

Tabella 2

La tabella 3 è la raffigurazione schematica di uno sprite visto da Amiga: notiamo che per ogni riga dello sprite sono assegnate due WORD e che ci sono delle configurazioni di cui non abbiamo ancora parlato.

Inutile dire che posizioneverticale, posizioneorizzontale, bitsdicontrollo e altezzadellosprite sono configurazioni a 8 bit poiché ognuna delle linee deve essere composta da non più di 16 bit. Adesso siamo veramente "a cavallo", basta definire una STRUTTURA di tipo SIMPLESPRITE attraverso la direttiva STRUCT del linguaggio C per creare il nostro sprite:

```
struct SimpleSprite imp0;
```

e, ovviamente, inserire i dati creati dalla nostra tabella nella struttura stessa: (notate che USHORT deve essere sempre scritto in maiuscolo e che per utilizzarlo si deve linkare (attaccare al programma) la libreria <exec/types.h> attraverso il comando INCLUDE <exec/types.h>)

```
USHORT sprite_data[]=
 {
  0,0,
  0x0248,0x0248,
  0x0248,0x0248,
  0x0248,0x0248,
  0x03f8,0x03f8,
  0x0c04,0x0c04,
  0x1203,0x1203,
  0x1203,0x1203,
  0x0c04,0x0c04,
  0x03f8,0x03f8,
  0x0248,0x0248,
  0x0248,0x0248,
  0x0248,0x0248,
  0,0
 };
```

16 BIT	
POSIZIONEVERTICALE	POSIZIONEORIZZONTALE
ALTEZZADELLOSPRITE	BITSDICONTROLLO(sono 8)
PAROLA 2 DELLA PRIMA LINEA	
PAROLA 1 DELLA PRIMA LINEA	
.	paia di parole che
.	definiscono il colore dello sprite
PAROLA 2 DELL' ULTIMA LINEA	
PAROLA 1 DELL' ULTIMA LINEA	
00000000	00000000
00000000	00000000

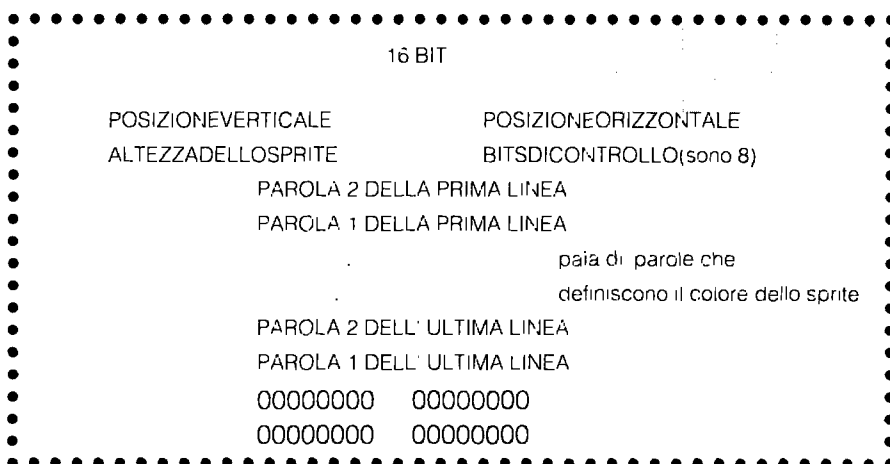



Tabella 3

I dati che abbiamo inserito rappresentano la forma di un insetto, essi sono preceduti dal prefisso 0x poiché questa è la rappresentazione dei numeri esadecimali in linguaggio C. Ora, creata la forma del nostro simpatico "scarafaggino", vediamo come visualizzarla sullo schermo: questo è possibile farlo attraverso la funzione ChangeSprite() che lega lo sprite alla ViewPort; senza entrare troppo nel dettaglio, la riga che segue "attacca" uno sprite allo schermo di Amiga e lo visualizza.

```
ChangeSprite(&Screen->ViewPort,&sprite,+&sprite_data);

MoveSprite(0,&sprite,sprite.x,
sprite.y);

WaitTOF();
```

L'ultimo comando forse vi sembrerà strano, WaitTOF() è una particolare funzione che dice al coprocessore che anima gli sprite di aspettare che il "videobeam", ossia il puntino luminoso che scorrendo compone l'immagine televisiva, abbia raggiunto il fondo dello schermo (serve ad evitare che ci siano sbavature grafiche a causa della scorretta apertura/chiusura dei canali di DMA).
Benissimo, non dimentichiamo di mettere dopo le condizioni di uscita dal programma il comando FreeSprite() che libera la memoria allocata per lo sprite specificato:

```
FreeSprite(sprite);
```

Ora siamo a posto, sappiamo come definire uno sprite, sappiamo come implementarlo e come visualizzarlo; sul disco di questo mese è dato un listato in linguaggio C (Atzec della Manx per la precisione) che visualizza un piccolo scarafaggio che cammina per lo schermo (ispirato da quello dell'Olivetti M20 che i nostalgici ricorderanno sicuramente).

Da C a Basic

Gestita la questione sprite semplici attraverso la programmazione in C, vediamo come affrontarla in Basic. Qualsiasi utente Amiga possiede il disco Extras fornito di base con la macchina. All'interno è presente una directory chiamata BASICDEMOS che contiene il programma OBJEDIT. Mandatelo in esecuzione e vi verrà chiesto se volete editare uno Sprite o un Bob (un Bob è simile ad uno sprite solo che può essere grande a piacere e avere fino a 32 colori ma, per contro, sono più lenti nei movimenti anche se sono gestiti dal Blitter). Data la poca precisione raggiungibile dallo sprite editor in dotazione, è sempre meglio creare, quando si programma in Basic, dei Bob. Il gioco è fatto, basta disegnare con il mouse la silhouette del vostro sprite e colorarlo a vostro piacere. Quando si è soddisfatti del proprio operato si può salvare il tutto affidandogli un nome.
La procedura che segue carica uno sprite precedentemente disegnato e lo posiziona alle coordinate 40,40.

```
'Apriamo il file esempio.spr che si trova nella directory sprite

OPEN"DF0:SPRITES/
ESEMPIO.SPR" FOR INPUT AS 1

' Stiamo assegnando allo sprite 1 la forma creata nel file di nome esempio.spr

OBJECT.SHAPE1,INPUT $(LOF(1),1)

' Adesso specifichiamo le coordinate orizzontali e verticali per poi visualizzare lo sprite.

OBJECT.X 1,40
OBJECT.Y 1,40
OBJECT.ON 1
```

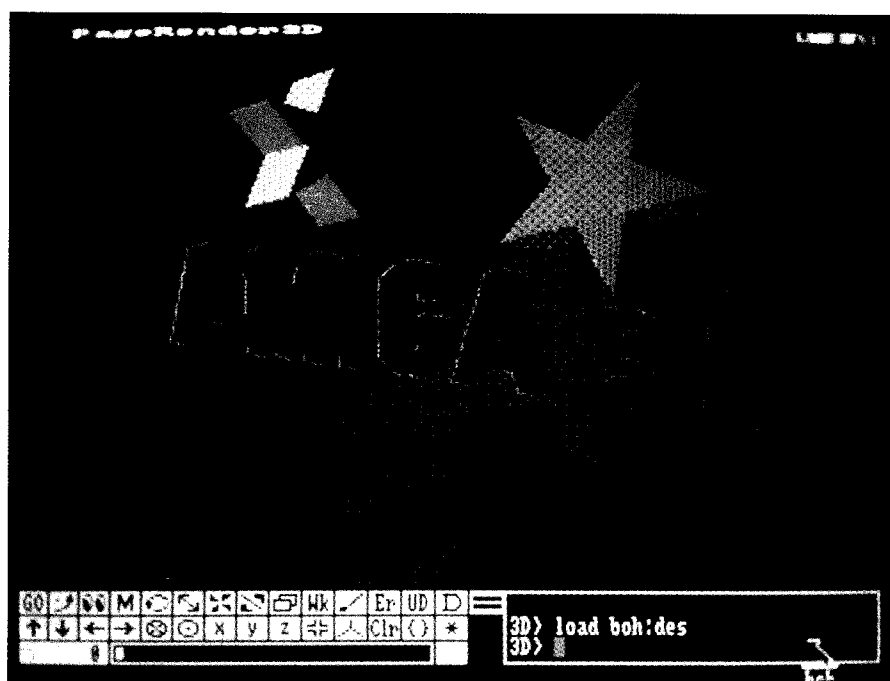
Le istruzioni, dettagliate, di utilizzo dello Sprite Editor sono presenti sul manuale dell'Amiga Basic.
Vediamo ora esempi di altre istruzioni messe a disposizione dal linguaggio Basic per la gestione di sprite e bob:

OBJECT.X 1,40 Assegna allo sprite 1 la coordinata x=40.
OBJECT.Y 1,50 Assegna allo sprite 1 la coordinata y=50.
OBJECT.VX 1,3 Assegna allo sprite 1 la velocità di 3 pollici al secondo in direzione orizzontale.
OBJECT.VY 2,5 Assegna allo sprite 2 la velocità di 5 pollici al secondo in direzione verticale.
OBJECT.AX 1,3 Assegna allo sprite 1 l'accelerazione di 3 pollici al secondo quadrato.
OBJECT.AY 1,3 Assegna allo sprite numero 1 l'accelerazione di 3 pollici al secondo quadrato.
OBJECT.ON 1 Visualizza lo sprite 1 sullo schermo
OBJECT.OFF 1 Cancella dallo schermo lo sprite 1
OBJECT.START 2 Fa muovere lo sprite secondo i parametri definiti da OBJECT.VX, OBJECT.VY, OBJECT.AX e OBJECT.AY.
OBJECT.STOP 2 Ferma lo sprite messo in movimento attraverso l'istruzione OBJECT.START
OBJECT.SHAPE 1,2 Assegna allo sprite 1 la sagoma dello sprite 2
OBJECT.CLOSE 1 Libera la memoria allocata dallo sprite 1
OBJECT.CLIP (1,1)-(100,100) Definisce un rettangolo di coordinate 1,1,100,100 e dice ad Amiga di non disegnare sprite al di fuori di quest'area.
OBJECT.PRIORITY 1,3 Assegna allo sprite numero 1 la priorità numero 3: questa istruzione ha effetto solo sui bob, la priorità più alta è 32727, la più bassa -32768.

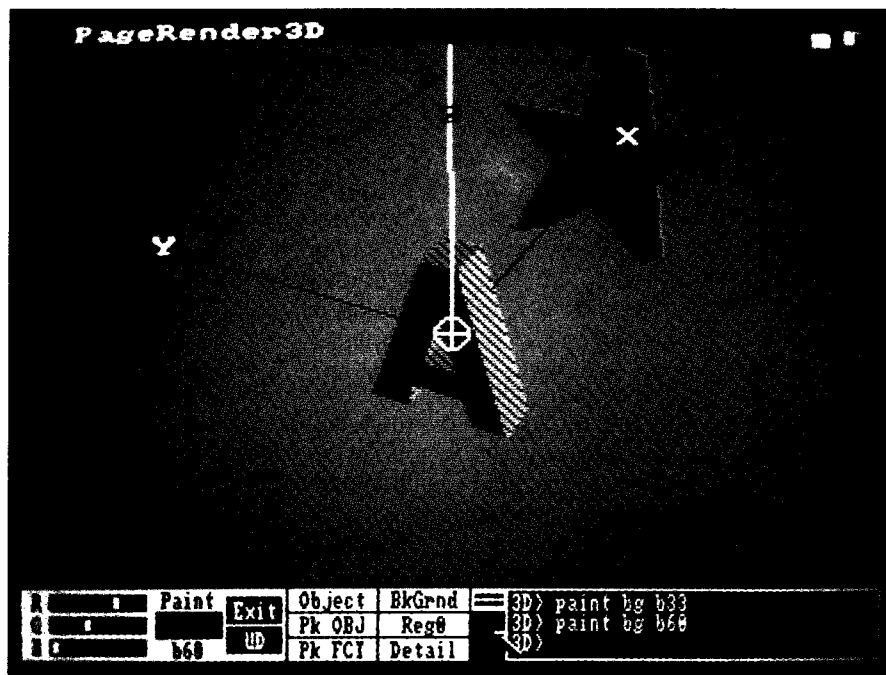
Come abbiamo visto il linguaggio Amiga Basic mette a disposizione una serie di comandi a volte anche molto potenti; grosso problema, però è la lentezza di esecuzione essendo il Basic un linguaggio interpretato.
Allo scopo di migliorare l'esecuzione dei vostri programmi, è consigliabile l'uso di un compilatore (ACBasic, per esempio) che trasforma il programma BASIC nell'equivalente in linguaggio macchina.
Attenzione:I programmi dimostrativi sono registrati sul disco nella directory utilities. ▲

PageRender 3D

Un programma molto versatile e ben studiato che sfrutta tutte le capacità grafiche e di animazione del favoloso Amiga.

Stefano Albarelli

La grafica tridimensionale computerizzata è un'arte che negli ultimi anni ha avuto enorme diffusione. Permette di visualizzare mondi immaginari e di animarli con grande efficacia, specialmente se per farlo si utilizzano computer molto potenti e veloci. Il nostro Amiga, nonostante non sia al livello di un Vax o di un Cray (tra i computer più potenti e costosi), con un software versatile e ben studiato, come lo può essere PageRender 3D, può darci molte soddisfazioni.
Gli autori del programma affermano che PageRender è stato preparato senza cercare idee e spunti da altri programmi già esistenti, ma prendendo in considerazione delle idee spesso originali che, in ben tre anni di prove e miglioramenti, hanno portato alla creazione di un programma veramente innovativo sotto molti aspetti e permette di sfruttare molto bene le capacità grafiche e di animazione di cui Amiga è dotato.

Il primo approccio

I programmatori della MindWare (la Software House che ha prodotto PageRender 3D) hanno dotato il loro programma di una insolita, ma molto pratica, interfaccia utente. E', infatti, possibile dare istruzioni al programma in tre modi: tramite i classici menu a tendina, per i comandi più semplici; tramite un menu a icone provvisto di uno slider per indicare il valore di eventuali parametri, o, infine, tramite una piccola finestra comandi posta affianco al menu a icone. Un particolare molto interessante e utile consiste in una grossa

finestra che l'utente può "sollevare" in qualsiasi momento (solitamente è visibile solo la drag-bar) spostando verso l'alto lo schermo del WorkBench (nel quale sono contenuti anche il menu a icone e la piccola finestra comandi). In tale finestra vengono trascritti tutti i comandi e i messaggi scaturiti dalla loro esecuzione, in modo tale da permettere all'utente di controllare in qualsiasi momento i comandi eseguiti in precedenza.

Lo spazio

Non appena caricato, si entra subito nel vivo del programma e sullo schermo viene visualizzato lo spazio, caratterizzato dai tre assi cartesiani e da quattro riquadri posti sul piano orizzontale; già da questo istante è possibile muoversi all'interno dello spazio mediante vari comandi che permettono di spostare la "camera" senza mutarne l'orientamento, di farla muovere attorno a uno degli assi tenendo l'obiettivo sempre puntato sul centro dello spazio o su di un oggetto, oppure di cambiare solamente la direzione dell'obiettivo senza spostare nello spazio la camera.
Una novità molto interessante consiste nel fatto che in PageRender 3D è possibile utilizzare tre diversi metodi per indicare coordinate e spostamenti: oltre alle solite coordinate cartesiane, è possibile utilizzare coordinate cilindriche, sferiche e tetraedriche, utili per particolari applicazioni. Naturalmente la "camera" ha anche un suo zoom, regolabile a piacimento per ottenere eventualmente anche l'effetto teleobiettivo o grandangolo.

Gli oggetti

PageRender 3D è uno di quei programmi che permettono di cominciare subito a lavorare, grazie ad un'ampia libreria di oggetti già pronti sul dischetto: solidi di vario tipo, lettere e numeri. Gli oggetti di questo programma sono considerati come un insieme di superfici piane. Un cubo, ad esempio, non è altro che un insieme di sei quadrati affiancati in un certo modo. Una volta caricati uno o più oggetti è possibile manipolarli a piacimento con una serie di comandi molto flessibili per deformarli, ingrandirli, rimpicciolirli, torcerli, adattarli a un piano sferico, cilindrico, conico o ondulato. Ogni oggetto può poi essere spostato nello spazio e colorato a piacimento (la minima unità colorabile è il singolo poligono che ne compone la superficie). Molto interessanti sono le opzioni che permettono di riprodurre più volte un singolo oggetto per creare un oggetto più complesso; se, ad esempio, abbiamo un oggetto a forma di stella, è possibile moltiplicarlo e creare una matrice tridimensionale formata da stelle.
L'oggetto può essere, inoltre, riprodotto in modo speculare rispetto a un piano, o in modo ciclico intorno ad un asse, creando un numero a piacere di oggetti. Se poi gli oggetti forniti assieme al programma non vi soddisfano e volete crearne di vostri, è possibile farlo mediante una sezione particolare del programma, dedicata esclusivamente alla creazione di oggetti.
Ora che abbiamo visto come agire

sullo scenario e sugli oggetti, vediamo come tutto ciò viene visualizzato sullo schermo.

La visualizzazione

PageRender 3D è un programma molto particolare nel suo genere in quanto tutte le operazioni di editing degli oggetti e dello scenario si compiono direttamente agendo sulla loro rappresentazione tridimensionale. Una miriade di opzioni permettono di scegliere la risoluzione dello schermo, il modo grafico e la modalità di visualizzazione degli oggetti; Pagerender 3D permette di utilizzare tutte le risoluzioni permesse da Amiga, il modo HAM e l'overscan.
Gli oggetti possono essere visualizzati in grid mode (ne vengono tracciati solo gli spigoli), in fill mode (che ne colora le superfici tenendo anche conto della sorgente di luce, ma colorando ogni faccia uniformemente), in outline mode (simile al grid mode con la differenza che gli spigoli sono del colore scelto per la faccia) e in ray tracing (una versione migliorata del fill mode che sfuma anche le singole facce in relazione alla posizione e all'intensità della sorgente di luce).
Una interessante opzione "dualscreen" permette di gestire contemporaneamente due schermi, in modo che ciò che deve essere visualizzato viene prima disegnato su di uno schermo nascosto che, una volta completo, va a sostituire lo schermo precedente.
Una nota di merito va ai programmatori di PageRender 3D per aver implementato la possibilità di poter davvero vedere la terza dimensione che manca allo schermo del nostro Amiga, e questo grazie ad un comando e a piccoli accessori aggiuntivi: il nome di questo comando è "stereo".
PageRender 3D permette di utilizzare due tecniche per poter vedere la terza dimensione: i classici occhialini colorati, con una lente rossa e una celeste, o un nuovo paio di occhiali progettati e prodotti dalla Haitex chiamati "X-Specs Glasses".
Il principio sul quale si basano gli occhiali con le lenti colorate consiste nel fatto che l'immagine riprodotta sullo schermo presenta in rosso l'immagine che un occhio deve vedere, e in celeste quella che deve vedere l'altro occhio; la lente rossa farà passare solo l'immagine celeste filtrando quella rossa, e viceversa, facendo in modo che i due occhi vedano così due immagini viste da due punti di vista differenti, distanti tra loro quanto i nostri due occhi.
Gli occhiali della Haitex, pur utilizzando lo stesso principio che consiste nel far vedere ai due occhi due immagini viste da punti di vista differenti, lo realizza in modo diverso: lo schermo viene portato in modo grafico interlacciato, e l'immagine di un occhio viene visualizzata sulle linee pari, quella dell'altro occhio sulle linee dispari.
Come forse saprete, quando lo schermo è in modo interlacciato, durante una passata del pennello elettronico sullo schermo vengono visualizzate solo le linee pari o quelle dispari (una passata quelle pari, la successiva quelle dispari, poi ancora quelle pari e così via); gli occhiali della Haitex hanno al posto delle lenti due pannelli a cristalli liquidi, che vengono alternativamente oscurati o resi trasparenti seguendo un segnale che arriva direttamente dal computer ogni volta che inizia la visualizzazione di un semiquadro.
In questo modo, durante la visualizzazione delle linee pari, l'occhio sinistro potrà vedere lo schermo, mentre l'occhio destro resterà al buio dietro al cristallo liquido annerito. Durante la visualizzazione delle linee dispari, l'occhio sinistro sarà all'oscuro, e l'occhio destro potrà vedere lo schermo. Risultato: l'occhio sinistro vede solo le linee pari (dove è presente la schermata dell'occhio sinistro), il destro le linee dispari (dove è presente la schermata dell'occhio destro).
I risultati migliori si ottengono sicuramente con gli occhiali a cristalli liquidi, perché rendono possibile la visione dei colori, e risultano di più semplice utilizzo rispetto a quelli con le lenti colorate, che oltretutto presentano il difetto di richiedere una buona taratura dei colori sullo schermo, pena l'annullamento dell'effetto di profondità, poiché il rosso deve avere la stessa tonalità della lente rossa per venire totalmente filtrato e così anche per il celeste.
Mentre gli occhialini colorati sono presenti nella confezione originale

del programma, gli occhiali a cristalli liquidi sono opzionali e in un futuro prossimo saranno disponibili anche sul mercato italiano. Altra caratteristica interessante del programma è la possibilità di utilizzare un Genlock per "fondere" le immagini ottenute con immagini televisive registrate o riprese in diretta.

Gli "script"

Fino adesso abbiamo visto come sia possibile creare uno scenario, inserirvi degli oggetti e posizionare la camera al suo interno per visualizzarne una particolare vista.
Ma le sorprese di PageRender 3D non sono ancora finite; tutte le operazioni necessarie per creare lo scenario possono essere eseguite direttamente passo dopo passo manualmente, oppure memorizzate in uno script, un file di tipo testo editabile anche mediante un normale Word Processor, o creabile direttamente mediante un'opzione "record" che permette di registrare le operazioni effettuate manualmente. Insieme al programma sono già presenti alcuni esempi di script che illustrano le possibili operazioni effettuabili con PageRender 3D.
Un'altra sorpresa consiste nel fatto che fino ad ora abbiamo trascurato una quarta dimensione che rende il tutto più interessante ed è presente anche nella nostra vita quotidiana, solo che in PageRender 3D è possibile muoversi al suo interno a piacimento; questa quarta dimensione è il tempo. PageRender 3D offre, infatti, la possibilità di creare vere e proprie animazioni salvando su di un file in formato IFF.anim una serie di fotogrammi, che è poi possibile rivedere con ShowAnim, un programma molto diffuso per animare sequenze di immagini. Tutto questo è possibile mediante uno specifico comando del PageRender che salva, in momenti ben precisi, la schermata. Vi sono molti script già pronti che salvano animazioni, ed esistono anche animazioni già pronte da vedere, per farsi subito un'idea delle potenzialità del programma.

(segue a pag. 80)

Top Ten

a cura del servizio Lago
Via Napoleona 16 22

	↔	Shadow of the Beast	Psygnosis
	▲	Dogs of War	Elite
	▲	Ghostbuster 2	Activision
4	Ⓔ	It Came from Desert	Cinemaware
5	Ⓔ	Space Ace	Ready Soft
6	Ⓔ	Future Wars	Delphine
7	▼	Indiana Jones	Lucas Film Games
8	▼	Batman, the Movie	Ocean
9	↔	Pro Tennis Tour	Ubi Soft
10	Ⓔ	Laser Squad	Reflex

a cura di Gianni Biagini

Nota dei voti

Grafica
Tiene conto della cura nel disegno dei fondali e degli sprite, dell'animazione, dello screen dei titoli.

Sonoro
Valuta la colonna sonora e gli effetti sonori.

Giocabilità
Tiene conto della varietà del gioco, delle opzioni e la possibilità di giocare in due e di eventuali scelte.

Durata
Indica per quanto tempo ci giocherete.

Generale
E' il voto generale dei valori precedenti e di tutto ciò che accompagna il gioco: confezione, manuale, gadget, ecc.

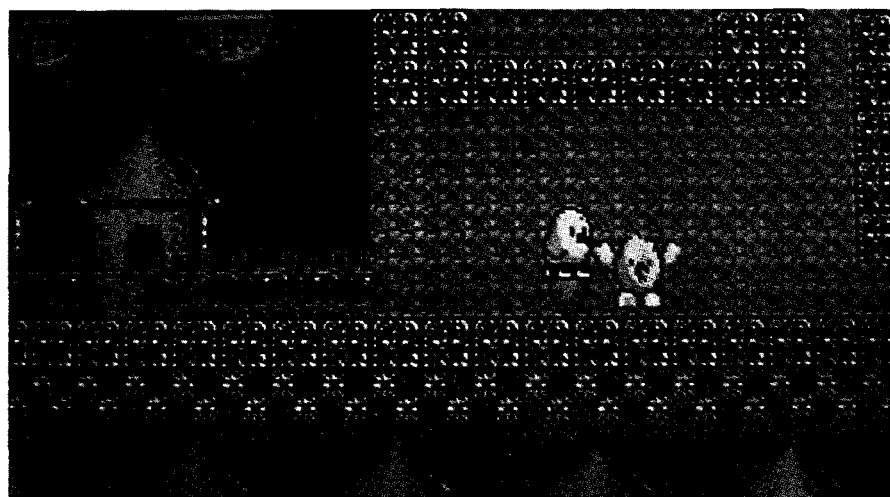
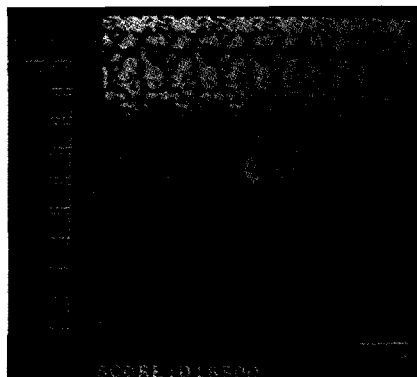
I voti vengono espressi in percentuale da 1 a 100.

The New Zealand Story

Grafica	79
Sonoro	70
Giocabilità	93
Durata	85
Generale	80

Software house: Ocean
Prezzo: 39.000 lire

Credo che nessuno di voi, soprattutto se appassionato di videogiochi, non abbia avuto modo di conoscere il "KIWI", quella specie di pulcino che la TAITO ha reso partecipe della storia della Nuova Zelanda. Sicuramente chi lo conosce sa che quantunque la grafica possa essere puerile (ma nel contempo molto nitida) la giocabilità è elevatissima; vi sono passaggi segreti da scoprire, un' infinità di opzioni, la possibilità di volare su palloncini, mongolfiere, dirigibili e chi più ne ha più ne metta. La base del gioco più o meno la si può rissumere in questi termini: un grosso e cattivo tricheco arriva a turbare i giochi vostri e dei vostri fratelli, voi riuscite a scappare dal sacco nel quale il vostro nemico ha messo gli altri KIWI e con indomito spirito di vendetta andate a liberarli. Le leggi della natura però sono inflessibili: preda o sarai predato, così il nostro pulcino si arma di arco e frecce e parte.
Sulla sua strada incontrerà altri abitanti della Nuova Zelanda che saranno quasi sempre ostili; vedrete piccoli lanciatori di boomerang, conchiglie che sputano bombe, lanciatori di giavellotto che cavalcano palloncini, mostri subacquei, balene e così via.
Allora non resta che rimboccarsi le maniche e partire all'esplorazione della Nuova Zelanda; ricordate però che, come ogni mondo fantastico che si rispetti, anche quello in cui combatterete contiene passaggi segreti che vi faranno avanzare di livello o prendere valide scorciatoie.
Tecnicamente il gioco è da considerarsi, a mio modesto parere, la miglior conversione di un gioco da bar che abbia mai visto, grazie anche alla stupenda prova della sempre migliore OCEAN, l'ubicazione dei passaggi segreti è identica, gli schermi sono ugualmente variopinti e nulla è lasciato al caso.
Forse si sarebbero potute apporre delle modifiche al sonoro che, come quello del bar, lascia a desiderare, comunque il gioco risulta veramente molto appassionante e vi farà passare bellissimi momenti.

New Zealand Story è distribuito da:
Leader Distribuzione
Via Mazzini, 15
Casciago (Va)
Tel. 0332/212255

Populous

Grafica	95
Sonoro	84
Giocabilità	88
Durata	90
Generale	90

Software house: Electronic Arts
Prezzo: 55.000 lire

Di giochi interessanti ne sono stati fatti tanti, ma stavolta la ECA si è superata, ha creato un simulatore di Dio !!!.
Forse la cosa potrà sembrare strana, eppure anche una divinità deve espletare alcune funzioni di routine. Dovrete, per esempio, controllare che la vostra orda di seguaci si espanda e proliferi nel migliore dei modi poiché più gente crede in voi, più potere acquisite, così da rimanere l'unico Dio a governare sulle terre del libro dei mondi. L'altro dio (non c'è niente da fare, il monoteismo va guadagnato) può essere guidato dal computer o essere incarnato da una persona mortale che possieda un Amiga e un modem o un cavetto di data-link (una seriale NULL-MODEM per intenderci). Molto divertente l'opzione fra due giocatori anche perché se non si è stabilito il contatto perfettamente si deve mandare un segnale di enquire il cui codice Hayek è ATE0 che, in un gioco di dei, ha molto poco a che fare. Capito lo spirito del gioco si devono acquisire le prime nozioni e, leggendo sul libretto delle istruzioni (in italiano) si perde una buona mezz'ora.
Sappiate che è possibile creare poli di attrazione religiosa per i fedeli, creare dei leader che verranno seguiti come profeti, creare ordini monastici, creare ordini cavallereschi, mandare i propri fedeli alla conquista di altri territori, mandarli a combattere per creare delle vere e proprie guerre religiose e, come ogni dio che si rispetti, modificare il corso della natura. Sono possibili: scosse telluriche, diluvi universali, eruzioni vulcaniche, apertura di paludi e modifica del terreno sottostante. Per dirla in breve voi assistete, e non solo da spettatore, allo sviluppo del mondo, vedrete i vostri sudditi creare nuovi insediamenti, andare alla ricerca di nuove terre, vedrete crescere il loro sviluppo bellico e alla fine potrete scatenare l' ARMAGEDDON ossia la battaglia finale che vi permetterà o meno di essere l'unica entità in grado di esser chiamata Dio, o di soccombere. Sicuramente Populous è un gioco interessantissimo, da giocarsi molto intelligentemente tenendo conto di ogni sottigliezza strategica, attenzione alle strategie troppo semplici: riempire il proprio

territorio di castelli, per esempio, dà una certa sicurezza nei confronti degli attacchi nemici ma, per contro, fino a che la rocca non è popolata, non ci saranno dei "camminatori" che andranno a colonizzare nuove terre, e quindi il vostro raggio di azione non si amplierà. Ricordate, inoltre, che è opportuno avere il maggior numero di camminatori anche perché è possibile fonderli tra loro per creare delle minuscole compagnie di fedeli; sono molto utili anche i cavalieri poiché come crociati vanno a combattere contro i fedeli dell'altro dio. Da tenere in considerazione lo stupendo sonoro che accompagna l'azione: una musica biblica da coro gregoriano segue le vostre teistiche imprese e un palpitare continuo vi indica la vicinanza al termine della gara.

The Promises Lands

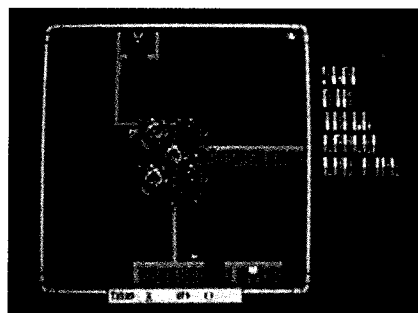
Oltre allo stupendo programma di cui sopra la ECA ha creato degli scenari che, in diverse epoche della storia dell'uomo, rendono più divertente giocare a POPULOUS. Questa volta possiamo vedere l'evolversi dell'uomo durante la Rivoluzione Francese, il selvaggio west, in un mondo di mattoncini da costruzione, e nella valle del silicio.
Graficamente questi scenari sono ben curati e possono essere cambiati durante la partita, da notare la presenza di ghigliottine sparse qua e là per i territori della rivoluzione.
In conclusione se avete POPULOUS acquistate anche gli scenari, sono molto carini e na vale la pena.

Populous e The Promises Lands sono distributi da: C.T.O. s.r.l.

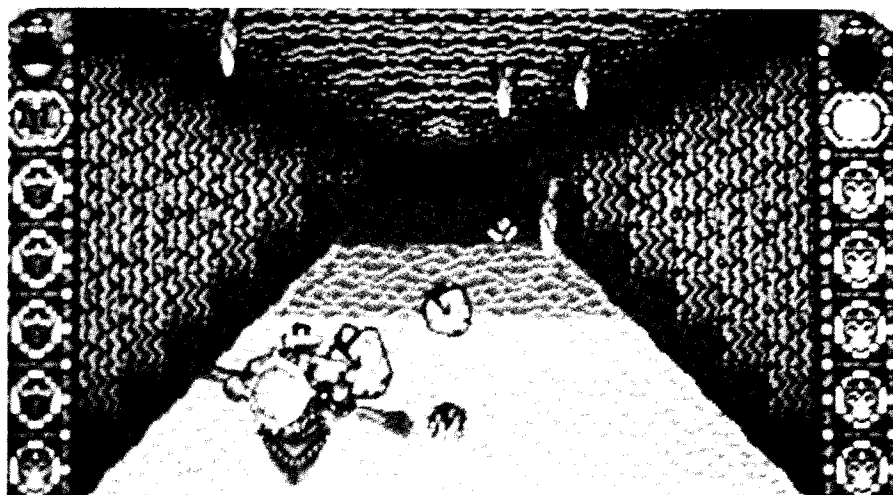
Laser Squad

Grafica	80
Sonoro	70
Giocabilità	85
Durata	85
Generale	86

Software house: Reflex
Prezzo: 39.000 lire

Laser Squad è uno di quei giochi che fa piacere che esistano poiché sono interessantissimi e rispettano l'intelligenza del videogiocatore; si tratta di un gioco di simulazione di una battaglia terroristica tra due squadre su uno scenario interattivo.
All'inizio dovete scegliere i tipi di corazza da assegnare ai vostri uomini e il livello di gioco del computer (che può lasciare il posto ad un vostro amico rendendo il tutto più avvincente)
A questo punto siete pronti per partire, attenzione perché il manuale è fondamentale (credo sia il miglior metodo per sconfiggere la pirateria quello di creare software di non intuitivo utilizzo).
A questo punto dovete veramente iniziare a leggere le istruzioni. Il gioco si utilizza a turno tra il primo e secondo giocatore, ognuno degli uomini della squadra ha uno specifico numero di punti azione che gli permettono di muoversi, sparare,

cercare oggetti, nascondere armi, perlustrare i più reconditi angoli della casa e così via.
E' disponibile per aiutarsi nelle proprie ricerche uno scanner al laser e vi posso assicurare che le opzioni di gioco sono veramente molte; il vostro scopo è quello di guadagnare dei punti vittoria che si acquisteranno provocando danni ai nemici.
Tecnicamente il gioco è molto ben gestito, l'azione è solamente strategica e sicuramente è più simile a un roll bar che a uno shoot'em-up e la grafica ricorda quella di un CAD.
Sicuramente, comunque, questo gioco vi appassionerà, previa lettura del manuale, però, ottima prova della BLADE (e della Reflex, che ha curato la traduzione in italiano del gioco e del manuale) che ha creato questa volta un programma di tutto rispetto.

Laser Squad è distribuito da:
Leader Distribuzione
Via Mazzini, 15 - Casciago (Va)
Tel. 0332/212255

Castle Warrior

Grafica	80
Sonoro	72
Giocabilità	81
Durata	75
Generale	80

Software house: Delphine
Prezzo: 29.000 lire

Ultima prova della DELPHINE, Software house francese. Questo programma, stando al manuale, dovrebbe essere fantastico.
Avete un giorno di tempo per liberare il vostro re dal veleno che lo sta uccidendo.
L'antidoto però si trova nel regno del temibilissimo Zandor che scatenerà i suoi mostri per farvi fallire nell'impresa. Ora tocca a voi!
Quantunque la trama richiami il filone di libri Fantasy ossia quei testi di spade, incantesimi e stregoni dove l'eroe è quasi sempre un barbaro dotato di forza sovrumana, il gioco si presenta decisamente inferiore alle aspettative, non c'è precisione nel movimento degli sprite, gli scroll anche se prospettici (voi siete so-

stanzialmente fermo e i corridoi scorrono ai vostri lati) sono molto lenti e poco avvincenti. Si unisca a ciò che è già stato detto la bassissima controllabilità dei personaggi e la lentezza in genere per capire quanto questo Castle Warrior sia un gioco fallito.
La Delphine ha cercato di dare un' immagine grafica piacevole al programma sfruttando (anche se poco) le potenzialità di Amiga: ne è risultato un prodotto poco intuitivo che non avvince quasi per niente. Come se non bastasse buona parte dell'azione si svolge evitando palline di fuoco che il mostro di turno getta per distruggervi. Insomma, una vera noia.
Questo è un gioco difficile da capire e poco divertente.

Castle Warrior è distribuito da:
Lago Softmail
Via Napoleona,16
22100 Como
Tel. 031/300174

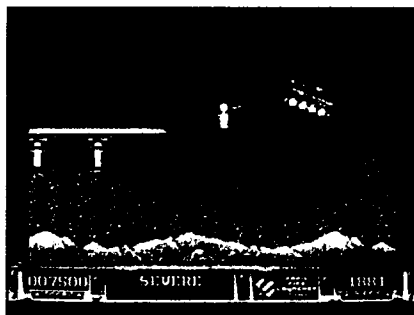
Forgotten Worlds

Grafica	94
Sonoro	85
Giocabilità	88
Durata	91
Generale	81

Software house: Capcom
Prezzo: 25.000 lire

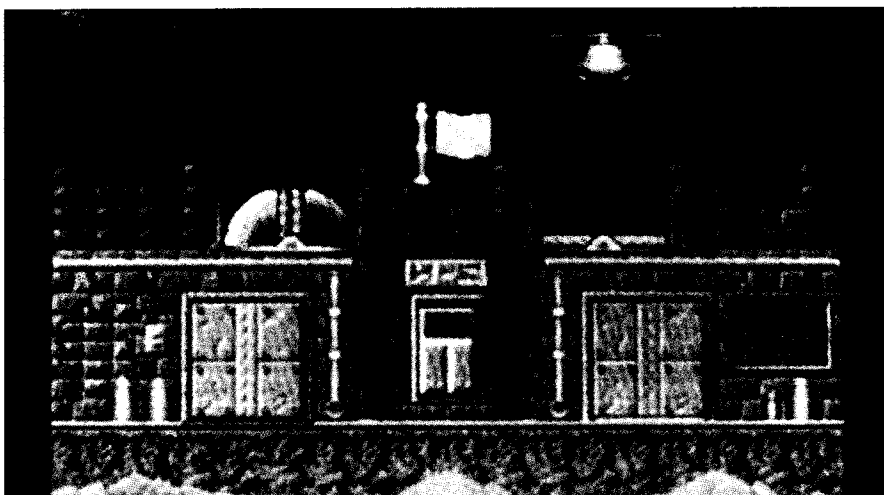
Già il nome lascia subito intuire che si parlerà di mondi dimenticati ed epoche postnucleari. Infatti l'imperatore BIOS che vuole distruggere tutta la civiltà, ha creato otto mostri invincibili per tutti tranne per gli uomini di Forgotten Worlds.
La CAPCOM è riuscita questa volta a creare una conversione dal gioco da bar originale fatta veramente bene, con gli scenari gestiti su due piani di scroll, e con un'azione veramente frenetica. Grosso neo del programma è la difficile controllabilità dei personaggi, poiché è possibile sparare in più angolazioni e, visto che si usa un Joystick a testa, all'inizio la gestione è un po' scomoda.
Tecnicamente il gioco si presenta

ben fatto, l'utilizzo degli scroll è molto fluido e piacevole, ricchissimi di particolari gli scenari sui quali si svolgono i combattimenti, ottime impressioni da parte della critica internazionale (soprattutto di Guida Videogiochi) e azione frenetica.
E' stato da molti definito come la conversione dell'anno (titolo che a mio parere spetta a New Zealand Story della Ocean), ed è molto interessante anche perché è possibile giocarlo in due persone il che accresce il divertimento e permette di non attendere interminabili minuti affinché il compagno "perda la vita" e vi lasci il turno.
Da segnalare oltrettutto la presenza di istruzioni in italiano per poter meglio capire trama e metodologie di gioco.
Ci stiamo avviando a sempre più precise conversioni degli arcade da bar; ultimamente (in particolar modo la OCEAN) tutte le principali Software House stanno puntando alla replica di successi già confermati, attenzione però a non mancare poi di originalità.

Forgotten Worlds è distribuito da:
Leader Distribuzione
Via Mazzini, 15 - Casciago (Va)
Tel. 0332/212255

Battle Valley

Grafica	88
Sonoro	74
Giocabilità	85
Durata	83
Generale	82

Software house: Hewson
Prezzo: 29.000 lire

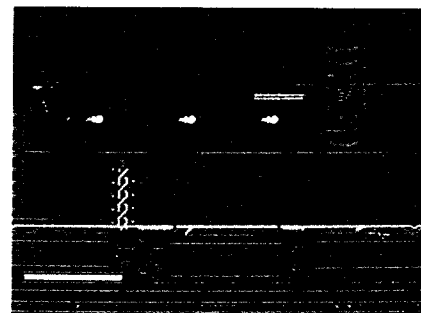
Come dice il manuale, il summit della pace ha avuto successo, tutti i missili vengono smantellati e le armi convenzionali distrutte. C'è però naturalmente un problema, un gruppo di terroristi che ha intenzione di distruggere il mondo occidentale ha rubato due tra i più sofisticati missili in circolazione: serve qualcuno, pronto a tutto, per riuscire a recuperare le armi rubate.
Questo qualcuno siete voi, noto mercenario ingaggiato dagli Stati Uniti. Le vostre forze a disposizione sono costituite da una serie di carri armati provvisti di uno sparo molto potente e di agili elicotteri dotati di un comodissimo gancio utile per ricostruire ponti per far passare i carri armati, che data la loro potenza di fuoco sono gli unici a poter distruggere la base dei terroristi e garantire quindi la pace nel mondo. Il gioco si presenta molto interessante sia dal punto di vista tecnico (lo scroll in parallasse su tre piani), sia da quello del divertimento: la grafica è molto ben curata soprattutto nelle sfumature di colore, scelta molto bene anche la palette; gli sprite, forse un po' semplici rispetto allo scenario, sono comunque ben definiti e, per quel che riguarda il carro armato, molto ben orientati poiché seguono la frastagliatura del terreno. Il sonoro non è all'altezza della grafica: sono presenti o musica o effetti speciali (non contemporaneamente) ma ambedue lasciano a desiderare.
Molto interessante comunque la varietà di nemici che incontrerete sul vostro cammino. In conclusione Battle Valley è un gioco piacevole e sicuramente divertente .

Battle Valley è distribuito da:
Leader Distribuzione

Dogs of War

Grafica	93
Sonoro	80
Giocabilità	84
Durata	85
Generale	84

Software house: Elite
Prezzo: 29.000 lire

La vita di un mercenario è piena di pericoli, ve lo insegna questo gioco della ELITE che, sullo stile di un IKARY WARRIORS un po' migliorato vi porterà in giro per il mondo alla ricerca di denaro, gloria e avventura.
Come già detto voi siete un militare professionista al quale vengono proposte 12 diverse missioni che, a seconda del grado di difficoltà, vi faranno guadagnare denaro e stima. Come ogni mercenario che si rispetti dovete portarvi dietro un vero e proprio arsenale per poter combattere tutti i tipi di nemici che incontrerete: potrete così avere lanciafiamme, pistole varie, cannoncini anticarro, fucili, mitragliatori e così via.
Attenzione in fase di acquisto del materiale bellico poiché per ogni arma si devono prendere anche adeguate serie di cartucce, altrimenti non le potrete utilizzare.
Molto carina la trovata della presen-

tazione delle missioni: vi muovete con il mouse su una cartina mondiale e in prossimità di ognuno dei punti dove potrete combattere compare il viso di chi comanda la missione, ovviamente con i tratti somatici del luogo. Per esempio, sul Golfo Persico vi è gente in turbante e in Cina un uomo con gli occhi a mandorla.
Sul piano tecnico il gioco si dimostra ben progettato, non ha niente o quasi più degli altri ed è carino da giocare, anche se un po' troppo difficile a volte.
Le armi sono ben presentate ma forse un po' scomode da cambiarsi, comunque la grafica è ben curata e ricca di particolari originali, ad esempio le Ferrari presenti sul percorso di guerra.
Altra nota simpatica e sicuramente positiva è il poter giocare in due o da soli.

Dogs of War è distribuito da:
Leader Distribuzione
Via Mazzini, 15 - Casciago (Va)
Tel. 0332/212255

Baal

Grafica	78
Sonoro	78
Giocabilità	82
Durata	76
Generale	80

Software house: Psyclapse
Prezzo: 29.000 lire

Ricordate Obliteretor? Fu uno dei primi spaziali della Psygnosis, la famosa Software house inglese che è sempre sulla cresta dell'onda e che produce sempre programmi innovativi e di una qualità superiore allo standard di mercato.
Quando qualche gioco rimane nella norma, non si evidenzia particolarmente, ecco la marca Psyclapse. Effettivamente con Obliteretor il nostro BAAL ha molto a che fare: si nota subito che l'azione è divisa in piani e che quando si uccide un nemico avviene la solita coreografica esplosione a forma di mostro accompagnata dal solito rumore metallico. La plancia comandi, invece, ricorda quella di Terrorpods, altro capolavoro della Psygnosis, però è molto più semplice da usare. Effettivamente un miscuglio di vecchi giochi non dovrebbe crearne uno

nuovo piacevole: BAAL risulta un programma senza infamia e senza particolare lode, quantunque le sue caratteristiche tecniche (riportate sulla confezione) parlino di scroll a 8 direzioni, 250 schermi dettagliati, 100 mostri diversi e 400 trappole sparse qua e là.
Oltre a quelle descritte dalla Psyclapse, però, ve ne sono altre abbastanza importanti per capire la qualità del gioco: il modo di camminare dell'omino è molto goffo e poco fluido, i nemici sparano forse un po' troppo, costringendovi ad attaccarli alle spalle per distruggerli e, infine, la capriola tipica di ogni gioco che permette di saltare i buchi nel terreno non riesce sempre. Concludendo il gioco in questione è graficamente ben curato, gli scenari sono molto vari ma forse un po' scontati, il sonoro, invece, è proprio "ridotto all'osso", gli unici effetti sono le esplosioni vostre e dei nemici. Invece, la giocabilità gli fa guadagnare dei punti poiché con l'utilizzo lo si conosce meglio e si riesce anche a divertirsi.

Baal è distribuito da:
Leader Distribuzione
Via Mazzini, 15 - Casciago (Va)
Tel. 0332/212255

I Servizi di

Amiga Magazine offre una serie di servizi per agevolare i propri lettori nel reperimento di software e materiale utile alla programmazione. E' disponibile l'intera libreria di dischetti di pubblico dominio curata da Fred Fish. Ogni dischetto contiene numerosi programmi e utility, spesso corredati da listati sorgenti e commenti degli autori.
Per districarsi fra le centinaia di programmi disponibili nei dischi di Fred Fish, come in precedenza su Transactor per Amiga, pubblicheremo nei prossimi mesi l'elenco ragionato.
Tale elenco riporta, divisi per categoria, tutti i programmi presenti, completandoli con informazioni quali la descrizione della funzione, l'autore, il numero di versione, la disponibilità del sorgente e il disco nel quale sono contenuti. I dischetti possono essere ordinati contrassegnando i numeri desiderati, purché la quantità sia di almeno 3 dischi.

BUONO D'ORDINE

Completare il buono d'ordine (o una sua fotocopia) e spedire in busta chiusa a:
I servizi di Amiga Magazine, Via Pola 9 - 20124 Milano

Si può allegare: assegno, contanti o fotocopia della ricevuta di versamento **c/c n.11666203** intestato a **Gruppo Editoriale Jackson.** Non si effettuano spedizioni in contrassegno.

Desidero ricevere i seguenti articoli; contrassegnare con una X i numeri di Fish Disk desiderati (minimo 3)

❑ 1	❑ 12	❑ 23	❑ 34	❑ 45	❑ 56	❑ 67	❑ 78	❑ 89	❑ 100	❑ 111	❑ 122	❑ 133	❑ 144	❑ 155	❑ 167
❑ 2	❑ 13	❑ 24	❑ 35	❑ 46	❑ 57	❑ 68	❑ 79	❑ 90	❑ 101	❑ 112	❑ 123	❑ 134	❑ 145	❑ 156	❑ 168
❑ 3	❑ 14	❑ 25	❑ 36	❑ 47	❑ 58	❑ 69	❑ 80	❑ 91	❑ 102	❑ 113	❑ 124	❑ 135	❑ 146	❑ 157	❑ 169
❑ 4	❑ 15	❑ 26	❑ 37	❑ 48	❑ 59	❑ 70	❑ 81	❑ 92	❑ 103	❑ 114	❑ 125	❑ 136	❑ 147	❑ 158	❑ 170
❑ 5	❑ 16	❑ 27	❑ 38	❑ 49	❑ 60	❑ 71	❑ 82	❑ 93	❑ 104	❑ 115	❑ 126	❑ 137	❑ 148	❑ 159	❑ 171
❑ 6	❑ 17	❑ 28	❑ 39	❑ 50	❑ 61	❑ 72	❑ 83	❑ 94	❑ 105	❑ 116	❑ 127	❑ 138	❑ 149	❑ 160	❑ 172
❑ 7	❑ 18	❑ 29	❑ 40	❑ 51	❑ 62	❑ 73	❑ 84	❑ 95	❑ 106	❑ 117	❑ 128	❑ 139	❑ 150	❑ 161	
❑ 8	❑ 19	❑ 30	❑ 41	❑ 52	❑ 63	❑ 74	❑ 85	❑ 96	❑ 107	❑ 118	❑ 129	❑ 140	❑ 151	❑ 162	
❑ 9	❑ 20	❑ 31	❑ 42	❑ 53	❑ 64	❑ 75	❑ 86	❑ 97	❑ 108	❑ 119	❑ 130	❑ 141	❑ 152	❑ 163	
❑ 10	❑ 21	❑ 32	❑ 43	❑ 54	❑ 65	❑ 76	❑ 87	❑ 98	❑ 109	❑ 120	❑ 131	❑ 142	❑ 153	❑ 165	
❑ 11	❑ 22	❑ 33	❑ 44	❑ 55	❑ 66	❑ 77	❑ 88	❑ 99	❑ 110	❑ 121	❑ 132	❑ 143	❑ 154	❑ 166	

Nota: il n° 164 non è disponibile

❑ 3 Fish Disk L. 24.000 (abbonati L.18.000)

❑ L. 5.000 per ogni disco aggiuntivo riferito ad ogni singolo ordine

Tutti i prezzi sono da intendersi IVA inclusa e spese di spedizione comprese.

Firma ____________________

(se minorenne quella di un genitore) Gli ordini non firmati non verranno evasi.

Cognome ____________________

Nome ____________________

Via ____________________

Cap __________ Città ____________________

Prov __________ Telefono ____________________

Codice abbonamento ____________________

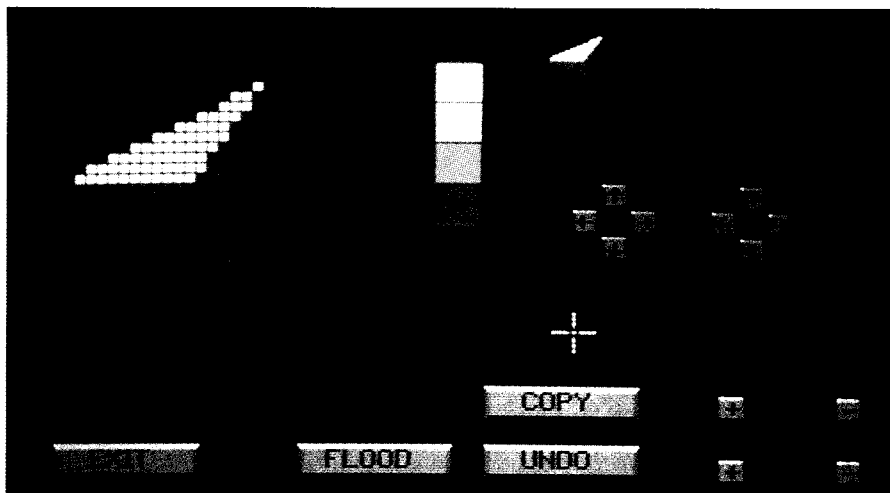
Shoot'Em-Up Construction Kit

Pronti a sbizzarrire la fantasia? Ecco,finalmente, un package per costruire in maniera semplice i videogiochi.

Daniele Cassanelli & Renato Acciardi

Con la realizzazione di questo programma la Outlaw intende riempire un settore del software molto fertile e redditizio come quello dei videogiochi che occupano una larga fetta della produzione software di Amiga. I videogame sono in parte una delle cause dell'iniziale successo di Amiga; basti pensare al primo glorioso gioco che lo rese popolare tra il pubblico, Defender Of The Crown, per arrivare agli ultimi "cartoni animati" molto più sofisticati e "bisognosi" di memoria come Dragon's Lair e Space Ace.
Questo programma si affianca ad altri dello stesso tipo con i quali è possibile realizzare altri tipi di videogame, come, ad esempio, le adventure. Una volta completato il gioco, sarà addirittura possibile realizzare versioni "stand alone" del gioco stesso, ossia ottenere versioni che funzioneranno indipendentemente dal SEUCK per essere utilizzati come giochi da distribuire ai nostri amici o, più ottimisticamente, da vendere a qualche Software House. Nella confezione troviamo due dischetti: l'Editor Disk e il Master Games Disk. Il primo contiene l'Editor ovvero il programma principale, mentre nel secondo troviamo tre giochi esempio creati in versione stand alone. Sempre nella confezione è presente un piccolo manuale, tradotto in italiano, che descrive brevemente tutte le opzioni dell'editor e una scheda di registrazione utenti presso la Lago SoftMail (ditta importatrice che ha tradotto il manuale) che permette di ricevere, eventuali, nuove release del programma. Il disco programma contenente l'Editor è stranamente protetto dalla copia per cui, pur essendo possibile duplicarne il contenuto, ogni volta che lo si vuole utilizzare è necessario inserire il disco originale pena l'impossibilità di salvare il lavoro realizzato.

L'Editor

Il programma è strutturato in modo abbastanza diverso dal resto dei

L'area di lavoro di Edit Sprite

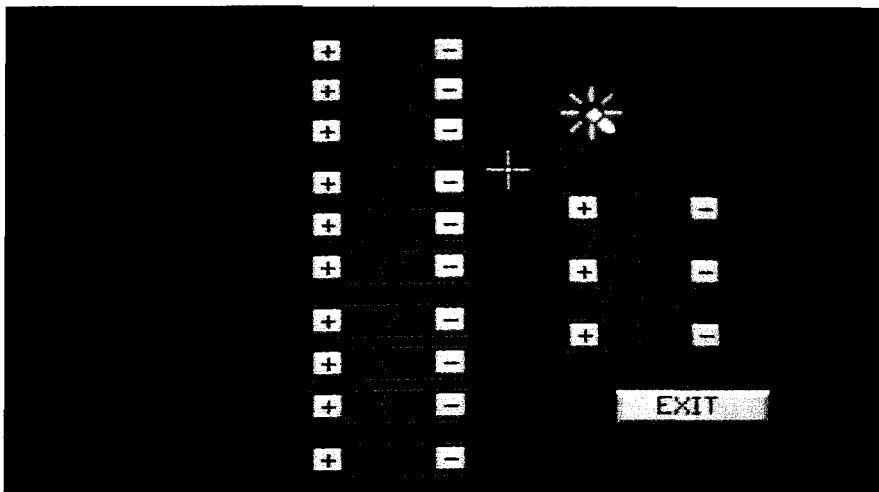
software per Amiga: non ci sono, infatti, i classici menu pull-down (a tendina) caratteristici di Amiga, ma al caricamento del programma ci troviamo di fronte ad un unico menu che si dirama in ben sette sottomenu attraverso i quali è possibile modificare gli sprite, i loro movimenti, lo sfondo, i suoni, i parametri dei giocatori, la posizione e le caratteristiche dei nemici. Tutte le voci del menu sono selezionabili sia con il mouse che con il joystick.
Il primo menu "Edit Sprite" permette di creare, modificare e cancellare i ben 99 sprite che, una volta animati, rappresenteranno gli oggetti del gioco. L'impostazione comune a questi sottomenu consiste in una griglia di 24 x 24 quadrati ossia l'ingrandimento dello sprite, il quale è presente, in dimensioni reali a fianco della griglia. Sempre a fianco della griglia troviamo la palette di otto colori, il numero dello sprite che si sta modificando e alcuni gadget tra cui il riempimento e l'effetto mirror (a specchio).
Attraverso il secondo menu (Edit Objects) è possibile animare gli sprite creati con il menu precedente facendo in modo che, ad esempio, ad ogni direzione del joystick corrisponda uno sprite, oppure che si abbia una animazione di 16 sprite. Sempre da questo sottomenu è possibile selezionare le caratteristiche dei nemici: il numero di colpi che possono subire, la frequenza e la direzione di sparo, la velocità, i punti che si ottengono colpendoli e gli effetti sonori che producono una volta colpiti.
Lo sfondo è il tema del terzo sottomenu: per una sua migliore gestione è stato diviso in 32 blocchi, ciascuno dei quali è modificabile in una griglia 32 x 32 che ricorda quella degli sprite. Dopo aver completato tutti blocchi è possibile "assemblare" il tutto per ottenere lo sfondo.
Argomento del quarto menu sono i suoni: da qui è possibile caricare e modificare i suoni (fortunatamente in formato IFF) in velocità e volume. Nel disco sono già presenti ben 72 suoni diversi che fanno fronte a tutte le esigenze, considerando che essendo modificabili possiamo agevolmente adottarli a nostro piacimento. La maggior parte delle caratteristiche e dei parametri del gioco la troviamo negli ultimi due menu (Edit Players Limitations e Edit Attack Waves) attraverso i quali si scelgono i movimenti dei nemici e il loro percorso. Sempre da questi menu si sceglie l'area in cui si può muovere il giocatore, la sua posizione iniziale e varie caratteristiche (direzione di sparo, velocità ecc.). Una caratteristica interessante riguarda la possibilità di unire più sprite nemici in uno solo. Infine c'è la possibilità di testare il gioco prima di memorizzarlo su disco, sia in modo normale (Proper Test) che con vite infinite (Cheat Test).

Con Edit Object è possibile creare animazioni con gli sprite

I parametri a disposizione del primo giocatore

Conclusioni

Sicuramente SEUCK avrà successo presso un pubblico giovane, visto e considerato che è un programma semplice da utilizzare. Concludiamo parlando del prezzo, che si colloca nella media del software di questo genere: 49.000 lire sono tutto sommato ragionevoli per un programma che sicuramente ci farà divertire.

SEUCK è in vendita presso: Lago SofMail - Via Napoleona, 16 22100 Como - Tel. 031/300174

Come utilizzare il dischetto

Prima di utilizzare il dischetto allegato alla rivista assicuratevi di averlo protetto dalla scrittura aprendone la finestrella, per evitare accidentali cancellazioni o contaminazioni da virus (il disco allegato è al 100% privo di virus ed il Gruppo Editoriale Jackson non si assume nessuna responsabilità in caso di contagio). A questo punto accendete il computer. Dopo la presentazione, dalla quale è possibile uscire premendo il tasto sinistro del mouse, verrà caricato il Workbench. Aprendo l'icona del disco (cliccando due volte con il tasto sinistro su di esso) appaiono i "cassetti" relativi a giochi, utilities, ecc. Aprite allo stesso modo il "cassetto" al quale siete interessati e caricate il programma desiderato. Alcuni programmi possono essere anche caricato da CLI con altre opzioni (vedere a parte le spiegazioni dei programmi). Per chi possiede Amiga con "soli" 512 K di memoria RAM è consigliabile chiudere tutte le finestre aperte tranne quella relativa al programma da far girare (che può essere eventualmente chiusa con un po' di prontezza di riflessi durante il caricamento). In tal modo il programma avrà a disposizione più memoria. In caso contrario, potrebbero verificarsi dei malfunzionamenti.
ATTENZIONE: prima di far partire ogni programma vi raccomandiamo di leggere attentamente le istruzioni del programma. I programmi Screen-Shift, KeyLock, Othello, DMouse e i programmi scherzo sono di pubblico dominio, e possono essere liberamente distribuiti. Inoltre, ricordiamo che, i programmi sorgenti ed eseguibili delle pagine di TRANSACTOR per AMIGA, si trovano nel cassetto TRANSACTOR e devono essere caricati da CLI. Tutti gli altri programmi presenti sul dischetto sono, invece, coperti da Copyright 1990 Gruppo Editoriale Jackson-Compute! Publications Inc. - Transactor for the Amiga U. K., e come tali non devono essere copiati e distribuiti senza la preventiva autorizzazione scritta dell'editore. Amiga Workbench 1.3 è copyright 1985, 1986,1987,1988 Commodore-Amiga Inc., tutti i diritti sono riservati. Per ogni questione tecnica riguardante il funzionamento del disco telefonare il Martedì e il Giovedì dalle ore 14.30 alle ore 16.00 al 02-6948287.

Come entrare in CLI

- Accendere il computer
- Inserire il disco con il Workbench
- Attendere il caricamento
- Cliccare due volte sul disco del Workbench
- Cliccare due volte sull'icona SHELL (per il nuovo CLI), oppure aprire il il cassetto System
- Selezionare l'icona CLI

Esempi di caricamento da CLI
- Entrare in CLI
- Digitare: COPY FROM C: TO RAM:
- Inserire il disco di Amiga Magazine e digitare CD DF0:
- Digitare CD UTILITIES

A questo punto, per esempio, si può digitare:

- DMouse
- Keylock

Amiga Magazine n. 11

SERVIZIO LETTORI **Compilare e spedire in busta chiusa a: GRUPPO EDITORIALE JACKSON Area Consumer - Via Pola, 9 - 20124 Milano**

A) Come giudichi questo numero di Amiga Magazine ?

❑ Ottimo
❑ Molto Buono
❑ Buono
❑ Discreto
❑ Sufficiente
❑ Insufficiente

B) Quale (i) articolo (i) o rubrica hai apprezzato di più?

Quale meno?

C) Cosa ti piacerebbe leggere nei prossimi numeri di Amiga Magazine ?

D) Ti è piaciuto il Software ?

E) Quante persone leggono la tua copia di Amiga Magazine ?

F) Possiedi un computer? Quale?

Quale (i) computer intendi acquistare in futuro?

G) Leggi altre riviste Jackson?
❑ SI ❑ NO

Quali?

H) Leggi altre riviste del settore?
❑ SI ❑ NO

Quali?

I) Oltre alle riviste dedicate al computer quali sono le tue letture preferite?

L) Quali sono i tuoi hobbies e maggiori interessi?

❑ Sport
❑ Musica
❑ Videoregistrazione
❑ Hi - Fi
❑ Fotografia
❑ Automobile
❑ Moto
❑ Viaggi

Nome
Cognome
Indirizzo
Età **Professione**
Città
Prov. **C.a.p.** **Tel.**

GRUPPO EDITORIALE JACKSON

LIBRI DI TESTO PER
E PROFE

Il Gruppo Editoriale Jackson, leader da molti anni nella editoria tecnica specializzata, da sempre inserito nell'area della formazione professionale con il più completo e vasto catalogo libri esistente sul mercato italiano, dal 1987 sta lavorando ad un ambizioso progetto editoriale legato al mondo della Scuola Secondaria Superiore, che vuole garantire una produzione editoriale didatticamente avanzata, completa ed aggiornata in quelle discipline e in quegli indirizzi di studio più direttamente legati al mondo dell'Informatica, dell'Elettronica e delle Telecomunicazioni.

Thomas L. Floyd
CIRCUITI ELETTRICI
Corso di elettrotecnica generale
Cod. SD667-pp. 750-L. 39.500

Pietro Adorni
ELETTROTECNICA GENERALE
Cod. SD607 - pp. 432 - L. 26.000

Dino Pellizzaro
MISURE ELETTRICHE
Cod. SD565-pp. 440-L. 27.000

Stephen J. Chapman
MACCHINE ELETTRICHE
Cod.: SE885-pp. 610-L. 42.000

Mauro Gargantini
FONDAMENTI DI ELETTRONICA ANALOGICA
Dispositivi, circuiti e applicazioni
Cod. SE769-pp. 896-L. 41.000

Ramakant A. Gayakwad
AMPLIFICATORI OPERAZIONALI E CIRCUITI INTEGRATI LINEARI
Cod. SE897-pp. 690-L. 39.000

Mauro Gargantini
Armando Zecchi
ELETTRONICA INTEGRATA LINEARE
Cod. SD490-pp. 392-L. 25.000

Herbert Taub
Donald Schilling
FONDAMENTI DI ELETTRONICA INTEGRATA DIGITALE
Cod. SD493-pp. 308-L. 26.000

Ronald J. Tocci
SISTEMI DIGITALI
Corso di elettronica digitale
Cod. SE890-pp. 980-L. 43.000

Pierfranco Ravotto
Eugenio Piana
PROGETTARE CON L'ELETTRONICA DIGITALE
Dalla logica cablata al programmabile
Cod. SD620-pp. 640-L. 38.000

Giuseppe Giuliano
MICROPROCESSORI
II Edizione
Architettura e programmazione
Cod. SE622-pp. 330-L. 25.000

Edward A. Maruggi
DISEGNO ELETTRONICO
Tecniche grafiche per la progettazione
Cod. SE645-pp. 368-L. 27.000

Fosco Bellomo
ELEMENTI PASSIVI TECNOLOGIE E DISPOSITIVI
Corso di tecnologie elettroniche - Vol. I
Cod. SD613-pp. 356-L. 27.000

Fosco Bellomo
ELEMENTI ATTIVI: TECNOLOGIE E DISPOSITIVI
Corso di tecnologie elettroniche Vol. II
Cod. SE892-pp. 380-L. 28.000

Fosco Bellomo
MICROELETTRONICA NUOVE TECNOLOGIE
Corso di tecnologie elettroniche - Vol. III
Cod. SD651-pp. 224-L. 21.000

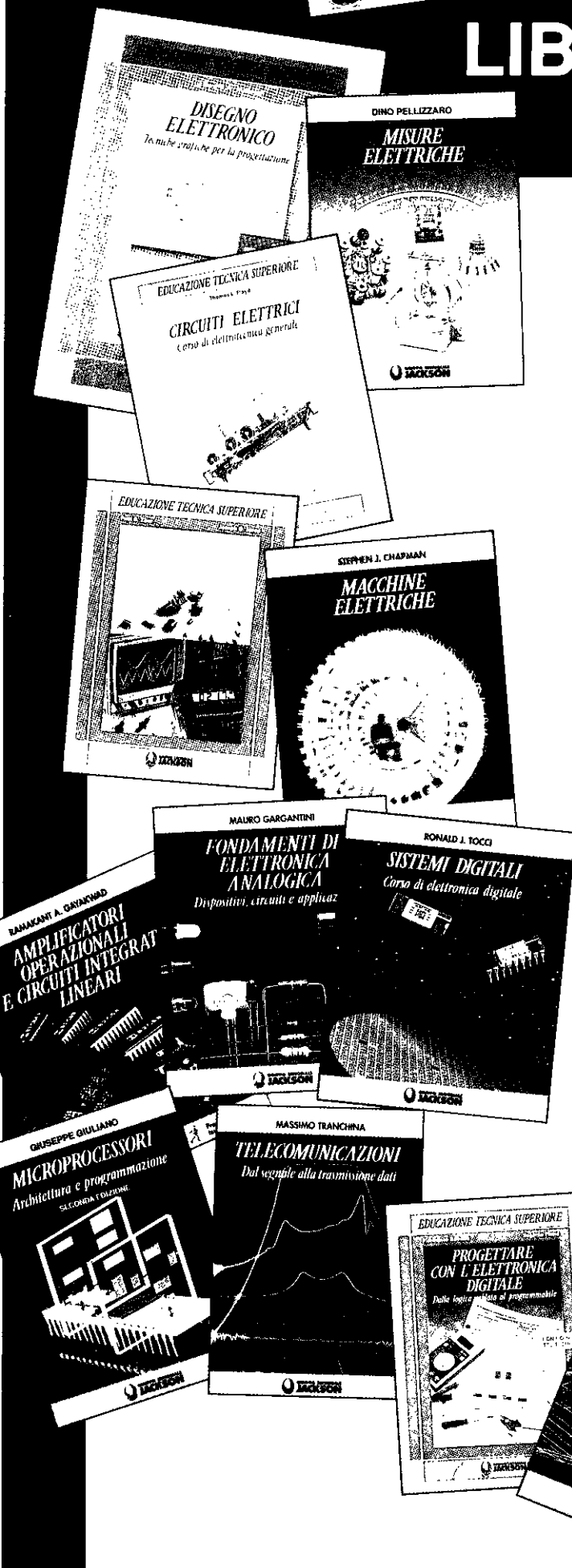
Renzo Traversini
MICROELETTRONICA: TECNOLOGIE E DISPOSITIVI
Cod. SD492-pp. 192-L. 20.000

Vincenzo Favale
ELEMENTI DI ELETTROTECNICA
Corso di elettronica Vol. I
Cod. SE895-pp. 354-L. 25.000

Vincenzo Favale
ELEMENTI DI ELETTRONICA LINEARE
Corso di elettronica Vol. II
Cod. SE755-pp. 350-L. 25.000

Vincenzo Favale
ELEMENTI DI ELETTRONICA DIGITALE E MICROPROCESSORI
Cod. SD754-pp. 476-L. 30.000

In vendita
le librerie s

ADEGUATI AI PROGETTI AMBRA ED ERGON

GLI ISTITUTI TECNICI SSIONALI

Paul B. Zbar
Joseph G. Sloop
LABORATORIO DI ELETTROTECNICA
Cod. SD497·pp. 296·L. 25.000

Paul B. Zbar
Joseph G. Sloop
LABORATORIO DI ELETTRONICA DI BASE
Cod. SD496·pp. 272·L. 22.500

Paul B. Zbar
Joseph G. Sloop
LABORATORIO DI ELETTRONICA INTEGRATA
Cod. SD495·pp. 246·L. 22.000

Paul B. Zbar
Joseph G. Sloop
DALL'ELETTRONICA ALL'ELETTRONICA INTEGRATA
Manuale di laboratorio
Cod. SD664·pp. 760·L. 48.000

Franco Sgorbani
Luciana Donelli
SISTEMI, AUTOMAZIONE E LABORATORIO
Cod. SE747·pp. 508·L. 39.000

Mario Malcangi
SISTEMI, MODELLI E PROCESSI
Corso di sistemi d'automazione - Vol. I
Cod. SD598·pp. 196·L. 20.000

Mario Malcangi
SISTEMI DIGITALI PER L'AUTOMAZIONE
Corso di sistemi d'automazione - Vol. II
Cod. SD599·pp. 248·L. 20.000

Mario Malcangi
SISTEMI, AUTOMAZIONE E CONTROLLO
Corso di sistemi d'automazione - Vol. III
Cod. SD491·pp. 192·L. 20.000

Felice Tarantini
TELEFONIA
Telefonia di base, sistemi di trasmissione, commutazione telefonica
Cod. SE666·pp. 622·L. 39.000

Ugo Sgubbi
Alessandro Gava
Giuseppe Saccardi
TELEMATICA
Corso teorico e pratico di trasmissione dati
Cod. SE879·pp. 474·L. 34.000

Paul H. Young
COMUNICAZIONI ELETTRICHE
Corso di radiolettronica
Cod. SD633·pp. 500·L. 37.000

Massimo Tranchina
TELECOMUNICAZIONI
Dal segnale alla trasmissione dati
Cod. SE869·pp. 280·L. 25.000

Peter Bishop
INFORMATICA GENERALE VOL. I
Cod. SE893·pp. 272·L. 20.000

Peter Bishop
INFORMATICA GENERALE VOL. II
Cod. SE894·pp. 334·L. 24.000

Mariangela Botti
Roberto Ranzani
DAL PROBLEMA AL PROGRAMMA
Cod. SD494·pp. 328·L. 26.000

Pierre Le Beux
PASCAL E TURBO PASCAL
II Edizione
Cod. SD775·pp. 384·L. 30.000

Pierre Le Beux
BASIC
Cod.: SD774·pp. 408·L. 30.000

Kip R. Irvine
COBOL
Cod. SE899·pp. 564·L. 42.000

Jack Purdum
C
Cod. SE898·pp. 456·L. 35.000

Roberto Doretti
Roberto Farabone
FORTRAN
II Edizione
Cod. SD770·pp. 388·L. 34.000

Salvatore Consentino
ORGANIZZAZIONE INDUSTRIALE STUDI DI FABBRICAZIONE E DISEGNO
Cod. SD624·pp. 220·L. 24.000

Nuovi strumenti per una scuola che cambia

in tutte
olastiche

Guida all'acquisto: non solo gioco

Caroline D. Hanlon

I giochi sono bellissimi, ma su Amiga si può trovare molto di più. Con l'avvento del videoediting da scrivania, Amiga sta guadagnandosi nuove schiere di adepti. Negli studi di videoproduzione, di registrazione, e perfino in certe università, Amiga sta guadagnando spazio sempre maggiore e sempre nuovo software viene alla luce, creando un virtuale labirinto di dischi, applicazioni, manuali e via dicendo. Per aiutarvi in una scelta sempre più ardua, abbiamo scelto alcuni programmi cosiddetti "professionali" da consigliarvi.

• Didattica

La didattica è dominio dei computer Apple e, in modo crescente, MS-DOS.
Grazie all'abbondanza di software e al basso costo di queste macchine, esse rimarranno probabilmente dominanti sul mercato scolastico. Amiga, comunque, si propone a livello universitario come sistema esperto interattivo e computer per applicazioni di intelligenza artificiale.
Probabilmente Amiga diventerà un giorno un sostituto del maestro, una macchina parlante e animata capace di interagire con gli studenti.

Mypaint
Centaur Software

Un programma di disegno e colorazione di immagini. Mypaint comprende 28 disegni da colorare, e un menu a icone per consentire ai bambini di creare i loro disegni. Per utilizzare il programma non è necessario saper leggere.

The Three Bears (Goldilocks)
Hilton-Android

Sesto programma della serie dei robot parlanti, The Three Bears è un aiuto ai bambini più piccoli che vogliono imparare parole nuove (in inglese, ovviamente). Il programma, interattivo, usa grafica e sintesi vocale per intrattenere i piccoli alunni mentre imparano.

• Applicazioni

Mentre esiste qualche programma studiato per l'utente casalingo, la maggior parte del software non ludico è orientato alla creatività e alla produttività. Nonostante Amiga sia un ottimo computer, il software in quest'area è ancora limitato. Man mano che Amiga conquista nuovi utenti non dediti solo ai giochi, dovremmo assistere alla nascita di nuovi prodotti nel campo.

Desktop Budget
Gold Disk

Gli utenti di questo programma, per la gestione di un bilancio familiare, possono utilizzarne la funzioni semplicemente, puntando il mouse e facendo click. Desktop Budget gestisce pagamenti mensili ed elabora i dati su scala annuale. Una calcolatrice aiuta a preparare i resoconti periodici, che possono essere stampati. E' presente anche un editor per la personalizzazione delle icone.

• Utility

Amiga è un ottimo computer per programmatori. Velocità, supporto di linguaggi, interfaccia utente e capacità di espansione di memoria danno ad Amiga un potenziale illimitato di risorse per programmare. Nuove utility, accessori per risparmiare tempo e aggiornamenti dei pacchetti più diffusi, mantengono l'area del software Amiga in continua crescita.

Aztec C Express 5.0, Developer Version
Manx Software Systems

La versione Developer contiene tutte le opzioni della versione Professional, più Z Editor, un text editor, VMake (per costruire grosse applicazioni), un assembler, le utility UniTools e SDB, un debugger completo.

Aztec C Express 5.0, Professional Version
Manx Software Systems

La versione 5.0 dell'Aztec C è un nuovo ambiente turbo-like completo di compilatore veloce che passa direttamente alla fase di assembly.
Il formato professionale include anche un editor basato su mouse, un project manager, linker, varie utility e SDB Jr., un debugger a livello di sorgente.

C.A.P.E. 68K Version 2.0
Innovatronics

La nuova versione di C.A.P.E. contiene un editor integrato, un ambiente di assemblaggio con file include precompilati, un'interfaccia ARexx e varie ottimizzazioni.

Devpac Amiga Version 2.0
MichTron

Assembler per Amiga. Devpac può lavorare a un ritmo di 70 mila linee di sorgente al minuto.
L'editor funziona sotto Intuition e può essere controllato sia da tastiera che da menu. Richiede Kickstart 1.2 o superiore.

ExpressCopy
Express-Way Software

ExpressCopy può eseguire backup o copia di file da hard a floppy disk alla velocità di un Mbyte al minuto, 600 Kbyte se è attiva l'opzione di verifica.
Possono venire creati nello stesso momento fino a quattro dischi di backup. Il programma è multitasking e funziona sia da CLI che da Workbench.

GFA-BASIC 3.0
Antic Software

Già disponibile in Europa, questo Basic offre interprete ad alta velocità, più di 300 comandi, inserimento di statement C e assembler, text editor incorporato, interprete runtime e accesso a librerie in linea.

HiSoft BASIC
MichTron

Un superset del PowerBasic.
HiSoft BASIC consente di scrivere accessori di scrivania (previa conoscenza del Gem) e routine di debug con il debugger Symbolic. Compila Amiga Basic ed è compatibile con il QuickBasic 3.

• Desktop publishing

Data per scontata la prevalenza di Macintosh, il desktop publishing su Amiga sta diventando sempre più diffuso, grazie alla facilità d'uso del computer e al prezzo assolutamente conveniente. Nei prossimi mesi vedremo applicazioni sempre più professionali, capaci, per esempio, di trattare con il colore a 24 bit, man mano che cresce presso gli sviluppatori l'attenzione verso questo settore del mercato.

Font-Works
ACS Software

Con questo editor e generatore di font, è possibile costruire famiglie di caratteri a colori, o catturare e modificare lettere da file IFF. Ogni font può accettare fino a 16 colori. Le opzioni includono ridimensionamento e modifiche in scala, color cycling, riempimento a pattern o a gradienti, tool per il disegno ed effetti speciali, accessibili via macro, come ombreggiatura, 3D, neon, bordatura, eccetera.

Media Line Clip Art Collection
Free Spirit

Questo disco contiene più di 200 immagini clip art e una utility che le mostra all'utente, per un'eventuale aggiunta a programmi per il desktop publishing che supportino la grafica IFF. I soggetti delle immagini spaziano dalle vacanze allo sport, dai computer ai paesaggi, e così via.

Media Line Font Disk 1 - 2
Free Spirit

Media Line Font Disks è una raccolta di font da utilizzare a discrezione dell'utente. Il disco 1 contiene nove font bitmap, mentre nel secondo le famiglie di caratteri sono vettoriali, per essere utilizzate in programmi 3D.

PageStream
Soft-Logik Publishing

PageStream non si ferma al semplice desktop publishing, ma offre opzioni a livello di fotocomposizione professionale come kerning avanzato e giustificazione verticale, oltre a formati multicolonna con flusso di testo automatico, ricerca e sostituzione, sillabazione automatica e manuale, correzione ortografica (peraltro, come la sillabazione, non utilissime in Italia se non a interpreti, traduttori, madrelingua...) e i tag, codifiche da inserire nel testo che attribuiscono ai paragrafi successivi formati e specifiche (corpo, carattere, attributi...) decisi dall'utente. Ancora, il programma è completo di dieci font, comandi per il layout della pagina, supporto di grafica a colori.

Profonts, volumi I-II
New Horizons

Compatibili con i programmi che utilizzano i font Amiga, questi pacchetti sono rivolti ai possessori di Amiga che si occupano di word processing e desktop publishing e possiedono una stampante a matrice di punti. Il primo volume contiene dieci famiglie di font dal look professionale, mentre il secondo è composto da 40 font di impatto, per l'uso creativo o per la titolazione.

• Desktop video e animazione

L'elaborazione video è la novità più recente del mercato Amiga. Spesso paragonata al desktop publishing per Macintosh, questa nuova tecnologia sta producendo lo stesso effetto su Amiga: diffonderlo in ambienti nuovi all'utilizzo del personal e al tempo stesso creare nuove figure professionali.
La videoelaborazione permette ad Amiga di fare il suo ingresso nel mondo del business, di solito più incline a misurarsi con l'MS-DOS o Macintosh, grazie alle sue doti ormai ben conosciute dai lettori di questa rivista. "Interfaccia personale tra l'artista e l'ambiente": ecco come è stata definito Amiga da uno dei maggiori produttori televisivi statunitensi.
Delle molte applicazioni possibili della videoelaborazione abbiamo cominciato a scoprire solo le più immediate, e possiamo aspettarci di vedere nel futuro prodotti innovativi e radicalmente diverse da tutto il software che oggi conosciamo.

ANIMagic
Aegis Development

ANIMagic è un editor di animazioni e un generatore di effetti speciali da usare con Videoscape 3D, Videotitler, DeluxePaint e altri programmi che lavorino con file IFF o ANIM. Si possono definire effetti personalizzati con una serie di strumenti piuttosto sofisticati a disposizione, ed editare a piacere le proprie animazioni.

Elan Performer
Elan Design

Elan Performer mette a portata di mouse il controllo totale delle animazioni in ogni programma che faccia uso del formato RIFF o ANIM. E' possibile assegnare i vari passi dell'animazione a singoli tasti, per poterli richiamare sullo schermo con estrema velocità.

LightBox
R & DL Productions

Con Lightbox è possibile creare cartoni animati utilizzando un programma di disegno. Le schermate possono succedersi a un ritmo di 30 al secondo. Richiede un Mbyte di RAM e il Kickstart 1.2 o 1.3.

Media Line Animation Backgrounds Disk
Free Spirit

Pacchetti come The Animator, Animate 3-D e DeluxeVideo possono avvalersi dei 30 diffe-

renti sfondi a colori per animazioni presenti su questi due dischi.
Gli sfondi sono disponibili in versione NTSC e PAL.

MovieSetter
Gold Disk

MovieSetter permette di creare sequenze di animazione di qualità professionale, grazie alle opzioni speciali del programma come successione ciclica dei colori, playback a 60 fotogrammi per secondo, guide lineari ed ellittiche, suono stereofonico, overscan e un display Wysiwyg.
I comandi vengono eseguiti puntando con il mouse i simboli presenti sullo schermo. Il programma include anche un file di personaggi, sempre di alta qualità, ed è possibile l'import da programmi di disegno.
E' necessario un Mbyte di RAM.

Turbo Silver
Impulse

Volete creare animazioni tridimensionali in ray-tracing?
Ecco Turbo Silver.
Con un limite di 512 fotogrammi per animazione, il programma funziona a 6-60 fotogrammi al secondo e può realizzare schermate ray-tracing in quattro minuti.
Supporta il genlock e l'overscan, ma richiede un hard disk, o un drive esterno, e una versione di Kickstart non precedente alla 1.2.

Viva
MichTron

Viva (il nome è acronimo di Visual Interfaced Video-Authoring system) permette di creare e gestire testo, grafica, suono, immagini video e animazioni, nonché videoregistratori, lettori di videodischi e qualunque altro dispositivo che concorra a creare un ambiente interattivo.
Il programma, con interfaccia a icone, è completo di editor, librerie grafiche, font e un modulo per la sintesi vocale.

• Software professionale

Si tratta di un settore tradizionalmente dominato dalla "Big Blu", ma Amiga ha già cominciato a infiltrarsi con successo nel campo e a ritagliarsi un piccolo ma significativo spazio. Molti credono che, per riuscire a sfondare negli ambienti professionali, Amiga debba proporsi con buone soluzioni di rete, per collegamenti a banche dati, servizi di posta elettronica e computer remoti. Altri, invece, pongono l'accento sulle notevoli capacità grafiche di Amiga, potenzialmente ideali per presentazioni che incorporino testo, immagini e suono. Man mano che aumenta la disponibilità di software realmente professionale per Amiga, il gioiello Commodore emergerà sempre più come alternativa a basso costo e alte prestazioni nei confronti di IBM e Mac.

Atredes 1.1
Incognito

La versione 1.1 di questo package per le telecomunicazioni include il pieno uso del mouse, il protocollo Z-modem, un programma per l'interazione on-line e menu concatenabili. Richiede un Mbyte di RAM.

Designer Database Series
Software Visions

I dischi della serie Designer Database contengono archivi e macro da usare con Microfiche Filer e Microfiche Filer Plus.
Il disco dedicato al mondo home offre dieci archivi già pronti, per videocassette, registrazioni audio, collezioni di francobolli, budget di casa e altro ancora. I dischi professionali contengono sette database, tra cui calendario, mail merge, analisi delle spese; è presente, inoltre, un set di macro.

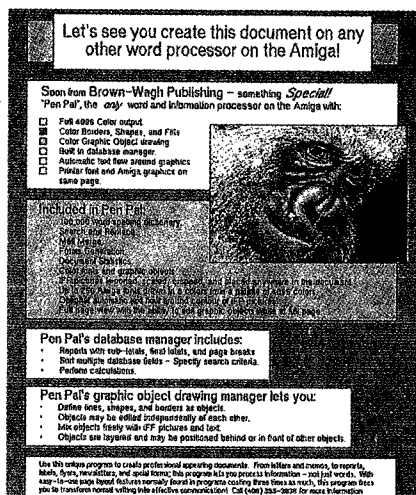
EasyLedgers
Brown-Wagh

EasyLedgers organizza i propri dati finanziari, in maniera facilitata dal costante utilizzo del mouse. Il manuale è ben fatto e contiene tutte le informazioni utili. Il pacchetto richiede un Mbyte di memoria ed è completo di file dimostrativi.

FreD
Gamma Software

FreD può memorizzare numeri di telefono, nomi e note di circa un milione di amici, colleghi e conoscenti, ed è in grado anche di comporre il numero di telefono al posto al posto dell'utente.
L'informazione viene posta in tre campi ognuno dei quali può contenere fino a 256 caratteri. Altre opzioni includono la ricerca di caratteri wildcard, accesso rapido all'editor di testo Notepad e il supporto per i file in ASCII e per ARexx. Il possesso di un modem è consigliato, ma non richiesto.

NagPlus 3.0
Gramma Software

NagPlus 3.0 è esattamente ciò che qualsiasi persona molto occupata può desiderare. Questo vero e proprio "gestore degli appuntamenti" può ricordare all'utente, per mezzo del monitor oppure a voce, fino a 99 appuntamenti, anniversari, compleanni, eccetera al giorno. NagPlus 3.0 può anche comporre automaticamente i numeri telefonici oppure stampare la lista delle attività in programma per la giornata.

Nimbus 1.3
Oxxi

Questa ultima versione del pacchetto finanziario prodotto da Oxxi per affari di dimensioni ridotte, presenta nuove opzioni come la produzione di moduli bianchi o prestampati, quantità decimali, una nuova procedure di chiusura dei conti di fine mese e un manuale aggiornato. Il programma gestisce clienti e venditori per nome ed è progettato per essere pronto da usare entro un'ora dall'apertura del pacchetto.

Pen Pal
Brown-Wagh

Pen Pal è un WP, archivio, e generatore di moduli per persone che devono produrre lettere e rapporti di lunghezza variabile da una a dieci pagine. Supporta un output di 4096 colori, immagini IFF, un dizionario di 100 mila parole, ricerca e riposizionamento, mail merge, un gestore di archivi incorporato, un modulo grafico, un generatore di moduli; e, infine, è in grado di fornire dati sul documento.

Project Master
Brown-Wagh

Con Project Master è possibile pianificare e gestire progetti di qualsiasi tipo utilizzando i sei moduli del pacchetto: Plan, Input, Statistics, Resources, Costs e Time.
Le opzioni a disposizione comprendono grafici in formato IFF e DrawPlus, inserimento dati in formato ASCII, analisi what-if, calcoli statistici e aggiornamento automatico delle date.

ProText
MichTron

Ultimissimo word processor della MichTron, comprende tutte le opzioni tipiche di un word processor professionale, compreso il controllo ortografico e un interprete di comandi.

Transcript
Gold Disk

Transcript è un word processor compatto, consistente in TransEdit, una versione più piccola del programma principale che manca delle funzioni di output, il programma stesso e TransSpell, un correttore ortografico di 90 mila parole.
Il programma crea automaticamente indici e mailmerge; all'interno dei menu quasi tutte le funzioni dei menu sono accompagnate da scorciatoie da tastiera.

• Musica

I musicisti professionisti guardano ad Amiga come a uno dei migliori computer per applicazioni musicali, per versatilità, purezza sonora e supporto dell'interfaccia MIDI.
La varietà di programmi musicali per Amiga consente di scegliere il software più adatto alle proprie esigenze e, nel futuro, la scelta sarà ancora più ampia, comprendendo anche applicazioni particolarmente orientate alla grafica, capaci di mostrare non solo note e pentagrammi, ma anche la musica vera e propria durante l'esecuzione.

A.M.A.S.
MicroDeal

Advanced MIDI Amiga Sampler, una combinazione hardware più software, permette di caricare, salvare ed elaborare suoni campionati.
Sono possibili campionamenti su uno e due canali, con analisi in tempo reale dello spettro. Il campionatore audio a otto bit è completo di interfaccia MIDI.

Dynamic Drums II
New Wave Software

Questa nuova release di Dynamic Drums presenta una nuova interfaccia utente, con nuove opzioni e un supporto MIDI potenziato, oltre a un nuovo editor a tutto schermo.
Il programma può essere personalizzato dal punto di vista grafico, con immagini in formato IFF. Sono disponibili, come opzioni, una interfaccia e una tastiera MIDI.

Dynamic Studio 3.0
New Wave Software

La versione 3.0 di questo programma offre miglioramenti nell'interfaccia utente, sincronizzazione SMPTE, editing più efficiente e un editor grafico per visualizzare le note.
Richiede un Mbyte di RAM, una tastiera e un'interfaccia MIDI.

EZ-CZ
New Wave Software

Programma dedicato alla tastiere Casio CZ, mostra tutti i parametri della tastiera su uno schermo apposito.
Dotato di numerose possibilità di editing, richiede una interfaccia MIDI e una tastiera Casio serie CZ.

Master Editor/Librarian
Sound Quest

La Sound Quest ha introdotto quattro nuovi editor/librerie per alcuni sintetizzatori tra cui M1, K1, D10/20/110 e DXII/TX802. Ogni editor/libreria possiede otto funzioni specifiche di editing dei banchi, editing globale, editing simultaneo di sintetizzatori multipli e schermi di help.

PageSync
Mindware International

Modulo da utilizzare con PageFlipper Plus F/X, PageSync permette la sincronizzazione delle animazioni create da PageFlipper con un dispositivo MIDI esterno.

Quest II: Texture
Sound Quest

La seconda versione di questo sequencer offre registrazione incrementale, editing animato, opzioni di quantizzazione e arpeggio, linearizzazione, scambio di accordi, canali MIDI multipli per traccia e un "panic button" che invia un comando speciale di cancellazione di tutte le note e il reset del controller.

Sonix SoundTrax Volumes One & Two
Aegis Development

Le canzoni contenute in questo package di due dischi sono tutte realizzate con suoni campionati e vanno impiegate all'interno di Aegis Sonix.

• Grafica

Che cosa ha decretato il successo di Amiga? La grafica, nei giochi come nel desktop publishing, nella videoelaborazione come nella musica.
I pacchetti grafici per Amiga stano diventando sempre maggiormente sofisticati.
Nel software professionale la grafica gioca un ruolo di primo piano. Ciò che prima richiedeva ore di tempo e costi spesso altissimi, oggi è realizzabile velocemente e con facilità.
La grafica continuerà a essere protagonista del mondo Amiga, e si introdurrà con sempre maggiore profondità anche nei settori più professionali.

ASDG-RESEP
ASDG

Utility progettata per l'uso con Scanlab Professional, ASDG-RESEP permette di rielaborare le immagini HAM create con Professional Page e inserirvi immagini a 24 bit per 16 milioni di colori.

DeluxePaint III
Electronic Arts

La terza versione di uno dei più noti software grafici per Amiga offre nuove opzioni: supporto dei modi a 64 colori ed Extra Half-Brite, overscan e animazione.
Il menu dei font è sostituito da un selettore, e il pacchetto comprende due dischi di immagini e animazioni.
C'è anche un programma per riprodurre animazioni create con DeluxePaint III senza bisogno di caricare l'intera applicazione. Si raccomanda un Mbyte di memoria.

Design 3-D
Gold Disk

Il nuovo programma di modellazione della Gold Disk può creare modelli wireframe con sei tipi di linee e 16 colori, con retinatura. Offre ombreggiatura veloce, estrusione degli assi e quattro viste degli oggetti, inclusa la prospettiva con quattro punti luce. Design-3D supporta anche testo tridimensionale, ed è completo di font e text editor.

Digi Paint 3

La versione 3 di Digi Paint presenta font sottoposti ad antialiasing, colorazione di superfici tridimensionali, nuovi strumenti per il disegno, software per l'image processing, disegno superbitmap e supporto dell'overscan.

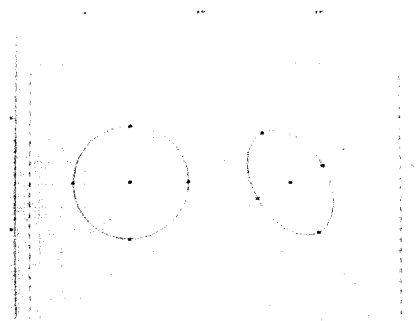
Icon Magic
Glacier Technologies

Questo programma di disegno di icone contiene tutto il necessario per disegnare e modificare icone personalizzate.
Dopo la creazione delle icone, il programma genera automaticamente il BASIC o il codice C o Assembler per i rispettivi bob, sprite o gadget. Icon Magic incorpora anche il formato ARexx-like Smartlink, per scambiare dati tra programmi differenti.

Optiks
Incognito Software

La caratteristica più interessante di questo programma per il ray-tracing è la sua compatibilità con il formato Targa a 24 bit e 16 milioni di colori, ma non mancano punti luce multipli e/o colorati e mapping di superfici. Richiede 512 Kbyte.

Professional Draw
Gold Disk

Professional Draw è un programma di disegno strutturato per oggetti colorati e progetti. Nel suo "pedigree" figurano una vasta gamma di strumenti per disegnare linee, poligoni e curve di Bezier, con spessori di linea definibili dall'utente, varietà di colori illimitata, macroistruzioni, opzioni di trasformazione come rotazione, distorsione e dimensionamento, stampa in alta risoluzione su stampante a punti e su stampante laser più un separatore di colori. E' compatibile con Professional Page 1.2.

Spritz
Glacier Technologies

Si tratta di un programma per utenti alle prime armi.
Contiene molte delle funzioni previste in software più costoso per usi professionali, oltre a comandi per "catturare" le schermate, un kit per creare diplomi e certificati di merito, supporto di 64 colori e compatibilità con Smartlink, la tecnologia ARexx-like che permette lo scambio di informazioni tra i programmi.

3D Options
Rainbow Edge Productions

Con 3D Options potete convertire le vostre immagini bitmap IFF in oggetti bi/tridimensionali, utilizzando tra l'altro opzioni di autotracing, riduzione dei colori e individuazione dei contorni.

X-Shell
Grafx Computing

X-Shell fornisce menu grafici personalizzati, da sostituire a quelli solo testo di X-CAD. Che è ovviamente necessario possedere.

(segue da pag. 62)

Page Render 3D

Durante la creazione i fotogrammi vengono lentamente visualizzati sullo schermo e salvati uno ad uno sul file anim, ma una volta terminata la creazione, mediante il programma ShowAnim, possono essere visualizzati ad una velocità di oltre 25 fotogrammi al secondo (ma anche più lentamente, nel caso sia necessario).
I risultati più interessanti si ottengono con animazioni che sfruttano l'effetto stereo degli occhiali: seguendo l'animazione sembra davvero di vedere degli oggetti reali che si muovono "dentro" il televisore, come se fosse una scatola vuota che li contiene.

Conclusioni

PageRender 3D, a nostro avviso, è un programma molto versatile che, con un po' di impegno, consente di raggiungere buoni risultati, specialmente sfruttando tecniche come l'animazione; unico piccolo neo, giustificabilissimo date le prestazioni grafiche, consiste nella necessità di almeno 1 MByte di memoria RAM, poiché già caricando il programma va "in fumo" la gran parte dei 512K di base.
PageRender 3D è un programma applicativo che risponde bene alle aspettative, soprattutto per quanto riguarda la sezione di visualizzazione.
Quindi se siete appassionati di animazione e grafica tridimensionale e avete voglia di concretizzare ciò che la vostra immaginazione vi suggerisce, PageRender 3D è il programma che fa per voi. ▲

PageRender 3D è in vendita presso: Pix Computer - Via F. D'Ovidio 6c 00137 Roma - Tel.06/8293507

COMPILARE E SPEDIRE IN BUSTA CHIUSA A: GRUPPO EDITORIALE JACKSON - VIA ROSELLINI 12 - 20124 MILANO

ABBONAMENTO RIVISTE JACKSON

ATTENZIONE La cedola di richiesta abbonamenti presenta un modulo speciale con una serie di domande a cui preghiamo vivamente di rispondere con precisione.

AVVERTENZA I rinnovi entreranno in vigore automaticamente a partire dal numero successivo alla data di scadenza dell'abbonamento precedente. Per motivi tecnici, i nuovi abbonamenti saranno attivati dopo 6 settimane dalla data di ricevimento della cedola di richiesta, o in data successiva nel caso di esplicita comunicazione da parte dell'abbonato.

❑ NUOVO ABBONAMENTO ❑ RINNOVO

Rivista	Numeri	Prezzo
❑ BIT	numeri 11	L. 53.000
❑ PC MAGAZINE	numeri 11	L. 52.000
❑ PC FLOPPY	numeri 11	L. 105.500
❑ INFORMATICA OGGI	numeri 11	L. 52.500
❑ INFORMATICA OGGI SETTIMANALE	numeri 40	L. 32.000
❑ TRASMISSIONE DATI E TELECOM.	numeri 11	L. 51.500
❑ COMPUTERGRAFICA & DTP	numeri 11	L. 51.000
❑ NTE COMPUSCUOLA	numeri 10	L. 33.000
❑ ELETTRONICA OGGI	numeri 20	L. 112.000
❑ AUTOMAZIONE OGGI	numeri 20	L. 80.000
❑ EO NEWS SETTIMANALE	numeri 40	L. 32.500
❑ STRUMENTAZIONE E MISURE OGGI	numeri 11	L. 53.500
❑ MECCANICA OGGI	numeri 11	L. 61.500
❑ STRUMENTI MUSICALI	numeri 11	L. 50.500
❑ WATT	numeri 20	L. 31.500
❑ FARE ELETTRONICA	numeri 12	L. 58.000
❑ AMIGA MAGAZINE (disk)	numeri 11	L. 123.500
❑ SUPERCOMMODORE (disk)	numeri 11	L. 110.000
❑ SUPERCOMMODORE (tape)	numeri 11	L. 75.000
❑ PC SOFTWARE 5 1/4	numeri 11	L. 106.000
❑ PC SOFTWARE 3 1/2	numeri 11	L. 132.000
❑ PC GAMES 5 1/4	numeri 11	L. 124.000
❑ PC GAMES 3 1/2	numeri 11	L. 132.500
❑ GUIDA VIDEOGIOCHI	numeri 11	L. 31.000

N.B. Per abbonamenti all'estero le tariffe dovranno essere raddoppiate. Non é prevista la spedizione via aerea

MODALITA' DI PAGAMENTO

Allego assegno n. ______ di L. ______

Banca ______

Ho effettuato versamento di L. ______ sul c/c postale n.11666203 intestato a Gruppo Editoriale Jackson-Milano e allego fotocopia della ricevuta.

Addebitate l'importo di L. ______ sulla carta di credito

AMERICAN EXPRESS ______

BANKAMERICARD ______

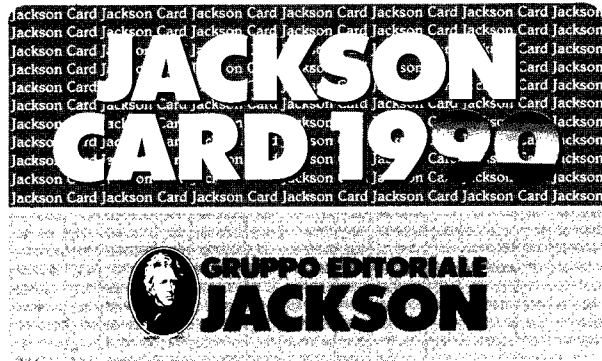
CARTA SI' ______

DINERS CLUB ______

Data di scadenza della carta di credito ______

Data ______ Firma ______

JACKSON CARD 1990

GRUPPO EDITORIALE JACKSON

Nuova, più ricca e sempre più preziosa

Ecco tutti i vantaggi che ti offre la Jackson Card 90: • sconti esclusivi e facilitazioni negli oltre 3000 punti di vendita convenzionati • sconto speciale del 10% sull'acquisto di libri Jackson • invio gratuito del catalogo libri 1990 e aggiornamento sulle ultime novità editoriali • invio gratuito della rivista Jackson Preview Magazine • buono acquisto speciale di L. 15.000 su ordini di libri effettuati per corrispondenza.

Ottenerla è facile.........

CEDOLA DI COMMISSIONE LIBRARIA

Si inviatemi i volumi sotto elencati:

❑ All'indirizzo privato ❑ All'indirizzo di lavoro

INDIRIZZO PRIVATO

COGNOME E NOME ______

VIA E NUMERO ______

CAP ______ CITTA' ______ PROV. ______

TEL. (______) ______ ANNO DI NASCITA 19 ______

TITOLO DI STUDIO: ❑ LAUREA ❑ MEDIA SUPERIORE ❑ MEDIA INFERIORE

INDIRIZZO LAVORO

DITTA O ENTE ______

VIA E NUMERO ______

CAP ______ CITTA' ______ PROV. ______

TEL. (______) ______ TELEX/FAX ______

INDICARE CHIARAMENTE CODICI E QUANTITA' DEI VOLUMI RICHIESTI

CODICE	Q.TA	CODICE	Q.TA	CODICE	Q.TA	CODICE	Q.TA

❑ Non sono titolare della Jackson Card

❑ Sono titolare della Jackson Card '90 n° ______ e ho quindi diritto allo sconto del 10% sino al 31/12/90

Per ricevere la Jackson Card '90 l'ordine minimo è di L.100.000.

Per ogni ordine verrà addebitato l'importo di L. 4.500 quale contributo fisso di spedizione.

MODALITA' DI PAGAMENTO

❑ Contro assegno postale ❑ Assegno allegato n° ______

Banca ______

❑ Versamento su c/c postale 11666203 a Voi intestato del quale allego fotocopia della ricevuta.

❑ Carta di credito ❑ American Express ❑ Visa ❑ Diner Club ❑ Carta Si

Autorizzo l'organizzazione sopra indicata ad addebitare l'importo sul mio Conto n° ______ Scadenza ______

❑ Richiedo l'emissione della fattura (formula riservata alle aziende) e comunico il numero di Partita I.V.A. ______

Data ______ Firma ______

COMPILARE E SPEDIRE IN BUSTA CHIUSA A: GRUPPO EDITORIALE JACKSON - VIA ROSELLINI 12 - 20124 MILANO

CEDOLA DI COMMISSIONE LIBRARIA

La cedola di commissione libraria presenta una serie di domande a cui preghiamo vivamente di rispondere con precisione. I dati serviranno a qualificare meglio le caratteristiche dei lettori per poter offrire un servizio informativo sulle nuove pubblicazioni e altre novità Jackson adeguato alle esigenze specifiche di ciascuno.

NUMERO DI DIPENDENTI
A ❑ da 1 a 49 **C** ❑ da 250 a 999
B ❑ da 50 a 249 **D** ❑ da 1000 in su

SETTORE AZIENDALE
A ❑ Acquisti
B ❑ Vendite
C ❑ Progettazione/Ricerca e Sviluppo
D ❑ Marketing e Comunicazione
E ❑ Produzione
F ❑ Amministrazione/Personale/ Finanza
G ❑ EDP
H ❑ Altro (specificare)

ATTIVITA' PRINCIPALE DELL'AZIENDA
P ❑ Produzione
D ❑ Distribuzione
S ❑ Servizi

SETTORE MERCEOLOGICO
A ❑ Informatica
B ❑ Automazione Industriale
C ❑ Meccanica
D ❑ Elettronica
E ❑ Strumentazione
F ❑ Elettricità e Energia
G ❑ Trasmissione Dati e Telecomunicazioni
H ❑ Finanza/Banche/Assicurazioni
I ❑ Editoria/Pubblicità/Comunicazione
L ❑ Pubblica Amministrazione Centrale/Locale
M ❑ Consulenza
N ❑ Istruzione (Scuola/Università)
O ❑ Altro (specificare)

POSIZIONE
A ❑ Alta Direzione
B ❑ EDP o Technical Manager
C ❑ Dirigente
D ❑ Tecnico/Progettista
E ❑ Consulente/Professionista
F ❑ Docente/Formatore
G ❑ Studente
H ❑ Altro (specificare)

INTERESSI PRINCIPALI
01 ❑ EDP
02 ❑ Personal Computer
03 ❑ Computer Grafica e Desktop Publishing
04 ❑ Trasmissione Dati e Reti
05 ❑ Home Computer e Videogiochi
06 ❑ Automazione Industriale
07 ❑ Meccanica
08 ❑ Strumentazione Elettronica
09 ❑ Telecomunicazioni e Telefonia
10 ❑ Elettronica Professionale
11 ❑ Elettronica Hobbistica
12 ❑ Elettrotecnica e Impianti Elettrici
13 ❑ Strumenti Musicali
14 ❑ Marketing e Management
15 ❑ Broadcast/Audio e Video Professionale
16 ❑ Didattica
17 ❑ Altro (specificare)

CHE PERSONAL COMPUTER POSSIEDE/UTILIZZA
DOS ❑ MS DOS, OS/2 e PC compatibili
MAC ❑ Macintosh
AMG ❑ Amiga
C64 ❑ Commodore 64
VAR ❑ Altro Home Computer (spec.)

...Abbonati alle riviste o acquista libri Jackson per almeno L. 100.000

Avere la Jackson Card é davvero facile: basta abbonarsi oppure acquistare libri Jackson per almento L.100.000 compilando le cedole che si trovano in questa pagina.Col primo numero del '90 di Jackson Preview Magazine ogni titolare riceverà la Jackson Card 90, l'elenco dei negozi convenzionati e lo speciale buono acquisto.

SERVIZIO QUALIFICAZIONE ABBONATI

ABBONAMENTO GRATUITO A 40 NUMERI, A SCELTA TRA LE SEGUENTI RIVISTE
❑ EO NEWS SETTIMANALE ❑ INFORMATICA OGGI SETTIMANALE
BARRARE LA CASELLA RELATIVA ALLA RIVISTA PRESCELTA

COGNOME E NOME ______
VIA È NUMERO ______
CAP ______ CITTA' ______ PROV ______
TEL. (______) ______ ANNO DI NASCITA 19 ______
TITOLO DI STUDIO: ❑ LAUREA ❑ MEDIA SUPERIORE ❑ MEDIA INFERIORE
NUMERO JACKSON CARD: GOLD ______ SILVER ______
DITTA O ENTE ______
VIA E NUMERO ______
CAP ______ CITTA' ______ PROV ______
TEL. (______) ______ TELEX/FAX ______

NUMERO DI DIPENDENTI
A ❑ da 1 a 49 **C** ❑ da 250 a 999
B ❑ da 50 a 249 **D** ❑ da 1000 in su

SETTORE AZIENDALE
A ❑ Acquisti
B ❑ Vendite
C ❑ Progettazione/Ricerca e Sviluppo
D ❑ Marketing e Comunicazione
E ❑ Produzione
F ❑ Amministrazione/Personale/ Finanza
G ❑ EDP
H ❑ Altro (specificare)

ATTIVITA' PRINCIPALE DELL'AZIENDA
P ❑ Produzione
D ❑ Distribuzione
S ❑ Servizi

SETTORE MERCEOLOGICO
A ❑ Informatica
B ❑ Automazione Industriale
C ❑ Meccanica
D ❑ Elettronica
E ❑ Strumentazione
F ❑ Elettricità e Energia
G ❑ Trasmissione Dati e Telecomunicazioni
H ❑ Finanza/Banche/Assicurazioni
I ❑ Editoria/Pubblicità/Comunicazione
L ❑ Pubblica Amministrazione Centrale/Locale
M ❑ Consulenza
N ❑ Istruzione (Scuola/Università)
O ❑ Altro (specificare)

POSIZIONE
A ❑ Alta Direzione
B ❑ EDP o Technical Manager
C ❑ Dirigente
D ❑ Tecnico/Progettista
E ❑ Consulente/Professionista
F ❑ Docente/Formatore
G ❑ Studente
H ❑ Altro (specificare)

INTERESSI PRINCIPALI
01 ❑ EDP
02 ❑ Personal Computer
03 ❑ Computer Grafica e Desktop Publishing
04 ❑ Trasmissione Dati e Reti
05 ❑ Home Computer e Videogiochi
06 ❑ Automazione Industriale
07 ❑ Meccanica
08 ❑ Strumentazione Elettronica
09 ❑ Telecomunicazioni e Telefonia
10 ❑ Elettronica Professionale
11 ❑ Elettronica Hobbistica
12 ❑ Elettrotecnica e Impianti Elettrici
13 ❑ Strumenti Musicali
14 ❑ Marketing e Management
15 ❑ Broadcast/Audio e Video Professionale
16 ❑ Didattica
17 ❑ Altro (specificare)

CHE PERSONAL COMPUTER POSSIEDE/UTILIZZA
DOS ❑ MS DOS, OS/2 e PC compatibili
MAC ❑ Macintosh
AMG ❑ Amiga
C64 ❑ Commodore 64
VAR ❑ Altro Home Computer (spec.)

GRUPPO EDITORIALE
JACKSON